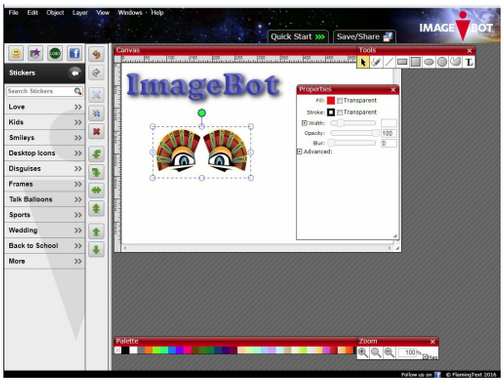
MAGAZYN

AMIGA NG

AmigaOS 4 - MorphOS - AROS (2) 01/2018

grafika wektorowa, droga do notebooka, mody do gry Aquaria, przejście z Amigi klasycznej na NG, i więcej...



WSTĘPNIAK

Miło znowu spotkać się, prawda? Mam nadzieję, że czekaliście z niecierpliwością, bo chcemy zbudować wokół siebie społeczność, taką jaka kiedyś była wokół Magazynu Amiga.

Wbrew pozorom tematów do opisywania nie brakuje. Poza stałą rubryką Nowości, która jest raczej kroniką, polecam dwa swoje artykuły. Pierwszy to przegląd programów do grafiki wektorowej pod AmigaOS 4; drugi jest o marzeniach amigowców o własnym notebooku, i wyboistej drodze do niego. Jest kolejny odcinek szkółki Hollywood. Fanów MorphOS-a może zainteresować tekst techniczny o współistnieniu tego systemu z MacOS X na jednym dysku – jak przeprowadzić instalację. Jest tekst o modach do gry Aquaria – pozwalają wydłużyć rozgrywkę, oferując niekiedy nowości względem oryginału. Wiele osób zainteresują wspomnienia – kolejny artykuł fana MorphOS-a opisuje jego drogę od Amigi 500 do MorphOS-a na Apple G5. Zaraz po nim jest tekst zakupowy o peryferiach do Apple G4 i G5 pod kątem wykorzystania pod MorphOS. Dalej, znów Hollywood, tym razem jak zabrać się za kompilacje aplikacji na Androida. AROS to opis narzędzia do odczytu partycji NTFS oraz recenzja gry Legend of Edgar. Po kącikach systemowych – tekst będący niejako uzupełnieniem artykułu wspominkowego. Napisany przez amigowca który, dla odmiany, nie był cały czas w środowisku, tylko wrócił do Amigi po latach – jak on widzi NG. Na koniec Humor i zapowiedzi artykułów do kolejnego wydania.

Numer jest zróżnicowany i myślę, że nie mniej ciekawy jak poprzedni.
Cóż teraz? Czołowi blogerzy napisali już swoje podsumowania roku, w starym 2017 już nic się nie wydarzy. Pozostaje nam życzyć sobie, aby ten nowy, 2018 rok, dostarczył prawdziwych powodów do świętowania.

Adam Mierzwa

Spis treści

Nowości	2
Grafika wektorowa i AmigaOS 4	4
Marzenia o laptopie	9
Hollywood #2 – E.M.M.A.	14
MorphOS i MacOS na jednym dysku	22
Aquaria – inne historie	28
Wspomnień czar starego dziadka – od Amigi do MorphOS-a	36
Prezenty dla PowerMac G4/G5	42
Hollywood APK Compiler	44
Dyski NTFS i AROS	48
Legenda Edgara	49
Galeria AROS-a	52
Powrót do Amigi po latach – moje doświadczenia z NG	54
Humor	58

MAGAZYN

AMIGA NG

Numer (2) 1/2018

Skład: PagePlus X9

Redakcja:
Adam Mierzwa
adammierzwa1@gmail.com

Autorzy tekstów:
Radosław Czernik
Konrad Czuba
Aleksander Giedyk
Robert Krajcarz
Jarosław Kuczyński
Adam Mierzwa
Rafał Pączko

Nowości

Koniec roku to gorący okres. Duże imprezy amigowe, a przede wszystkim ostateczny termin realizacji obiecanych rzeczy na dany rok. Jak bardzo ten czas spełnił nasze oczekiwania?

CodeBench

Po czterech latach ciszy ukazała się nowa wersja tego zintegrowanego środowiska programistycznego. Wymagany jest AmigaOS w wersji minimum 4.1 Update 1 (rekomendowany minimum Update 3). Są dwie wersje pakietu – dla SDK (stanowi jego dopełnienie) i Hollywood (wymagana wersja 4 bądź wyższa).
Skok numeracji Codebench to 0.42 –> 0.55. Dokładna lista zmian znajduje się w dokumentacji programu.
Obie wersje (SDK i Hollywood) można pobrać stąd:
codebench.co.uk/downloads.html

AmiWest 2017

Wbrew oczekiwaniom nie było premiery A1222. Potwierdzono że trwają prace nad sterownikami sieci i dźwięku oraz optymalizacją FPU. Zaprezentowano Tabora publicznie z aktualną wersja AmigaOS dla niego. Nie było też LibreOffice ani nowych czy odnowionych programów.
Zapowiedziano Enhancer 2.0, który ma zawierać nowe sterowniki RadeonHD. Mają one obsługiwać karty z serii Polaris oraz udostępniać całą pamięć karty graficznej. Nowy Enhancer ma też mieć narzędzia do archiwizacji i backupu.
Po raz kolejny pokazano laptop A.L.I.C.E., tym razem na innej konfiguracji laptopa (też Acer). Ma być dostępny w sprzedaży jeszcze w tym roku.
Zaprezentowano MorphOS w wersji 3.10 na AmigaOne X5000.
Tradycyjnie najważniejsze wydarzenie dla fanów AmigaOS 4 w tym roku dla większości było rozczarowaniem.

Amiga 32

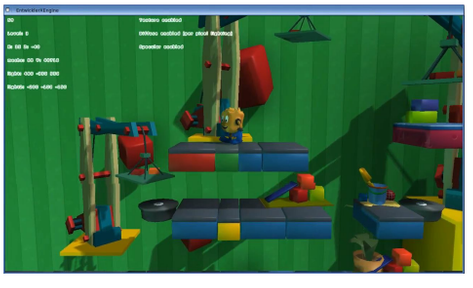
Niespełna tydzień po zakończeniu AmiWest, w Neuss (Niemcy) odbyła się kolejne ważna impreza. Od edycji 30 pomału wyrasta ona na najważniejsze wydarzenie dla Amigi nie tylko w Europie ale i na świecie.

Premierę miała gra „Wings! Remastered Edition", były pomniejsze rzeczy typu nowa wersja Amiga Racer, ale największym zainteresowaniem cieszył się Vampire.
Większość tego co pokazano była wtórna wobec AmiWest, bądź dotyczyła klasycznej Amigi.
Tak jak wielu przypuszczało phase 5 wbrew zapowiedziom nie zaprezentowała żadnego działającego produktu. Można było tylko obejrzeć płytkę Blizzard 1260 MK3 i kilka innych prototypowych, niczego w działaniu.

Rzetelna relacja jest tutaj:
amigax5000.blog/2017/10/29/amiga-32/

Spencer

Grupa Entwickler-X udostępniła roboczą wersję gry Spencer.
Plik można pobrać stąd:
www.entwickler-x.de/zeugs/Spencer.lha

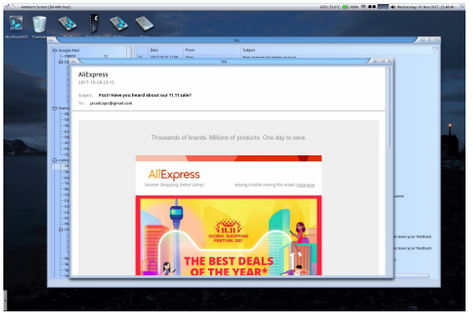


Dla najlepszej wydajności wymagane są Warp3D Nova oraz OpenGL ES 2, ale możliwe jest też działanie na słabszych konfiguracjach.
Pomocny może być wątek autora programu:
os4welt.de/viewtopic.php?f=8&t=2007

Iris

Pojawiły się pierwsze konkrety dotyczące nowego klienta e-mail dla systemu MorphOS. Autorem jest Jacek Piszczek.
Jak widać na zrzucie obsługa HTML w mejlach wygląda dobrze.
Nie jest planowana obsługa Exchange (na szczęście większość serwerów Exchange ma frontend IMAP), ani

czytnika grup dyskusyjnych. Być może będzie obsługa Web/CalDAV.



Prawdopodobnie Iris znajdzie się w wersji 3.12 MorphOS.

A.L.I.C.E.

Po AmiWest ten, uśpiony zdawałoby się, projekt objawił się namacalnymi konkretami.
Robert Bernardo zaprezentował wersję przedsprzedażową, niebrandowaną; jest to konfiguracja oparta o laptop Acer E5-575:
blog.retrolink.com/2017/10/unboxing-alice-laptop-incorporating.html

Rabbit Hole computing

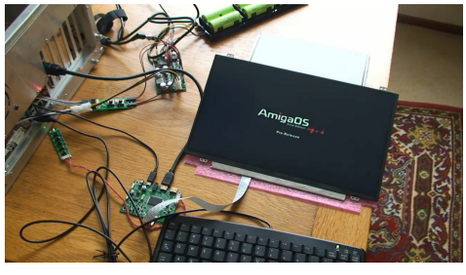
Podał też stronę internetową Rabbit Hole Computing:
www.rabbitholecomputing.com/hardware

Można tam zobaczyć aż trzy konfiguracje – pozostałe dwie to laptop Dell Latitude 5280 oraz nettop Gigabyte BRIX i3-6100U.

Sam zakup następuje po przekierowaniu na stronę firmy Inertial Computing, która podaje jako swe motto „Keeping Yesterday Alive”. Można u nich kupić inne drobne peryferia typu przejściówki SCSI2SD.
Całość przedsięwzięcia sprawia wrażenie improwizacji.

PORTIA

Skrót ten pochodzi od **P**ortable **T**abor **I**nspired **A**migaOne A1222.
Projekt „zrób to sam”, którego autorem jest Hans de Ruiter, znany z tworzenia sterowników RadeonHD. Jak wynika z rozwinięcia skrótu – celem jest budowa laptopa na bazie płyty Tabor.



Nie jest to pierwsza próba zbudowania notebooka w oparciu o płytę PPC, wystarczy przywołać projekt Novena. Więcej – na blogu autora:
keasigmadelta.com/store/diy-laptop-landing/

Notebook od ACube

Roberto Innocenti, który próbuje przebić się ze swoim laptopem na otwartej licencji już od kilku lat, w tym roku wreszcie zdecydował się na śmiałe i owocne działania. Powstało stowarzyszenie „Power Progress Community”, które prowadzi kampanię zbiórkową mającą na celu doprowadzenie do szczęśliwego finału. Zamiarem jest przerobienie płyty referencyjnej T2080 w taki sposób, by dała się zamknąć w typowej obudowie laptopa 15 cali.

Przeróbki płyty ma dokonać znana nam firma ACube.
Prace podzielono na pięć etapów, które otwierane są w miarę napływania funduszy.
Całość ma kosztować 62.250 euro. W tej cenie ma być przygotowane 5 płyt prototypowych, ich przetestowanie, stworzenie dokumentacji i wstępna certyfikacja CE.

Póki co zaprezentowano schemat blokowy. Postęp można śledzić na stronie projektu:
www.powerpc-notebook.org/en/

RNOWidgets

Widżety (niewielkie programy na pulpit) dla systemu MorphOS.
W zestawie zegar analogowy/cyfrowy, czytnik RSS, kalendarz, „żółta karteczka” oraz ramka na zdjęcia. Wszystkie widżety są skalowalne, obsługują przezroczystość, jest obsługa AREXX-a.
Pakiet dostarczył ‘jPV^RNO’ i można go pobrać ze strony projektu jpv.wmhost.com/jpv_software/RNOWidgets/
Także z MorphOS Storage, Aminetu oraz za pomocą Gruncha.

Jak widać, ostatni kwartał i okres bożonarodzeniowy nie zaskoczył nas niczym przełomowym w temacie Amigi NG.

Grafika wektorowa i AmigaOS 4

truman burbank

Grafika wektorowa zawsze była mniej popularna od rastrowej. Zawsze był mniejszy wybór programów, z prostej przyczyny – więcej ludzi potrzebuje obrobić zdjęcia, niż rysować. I to jak rysować – grafika wektorowa nie jest typowym rysowaniem. Wymaga precyzji, i dla osób z artystyczną duszą może wydawać się ograniczająca. Przy pierwszym kontakcie nie daje tej swobody, co np. malowanie.

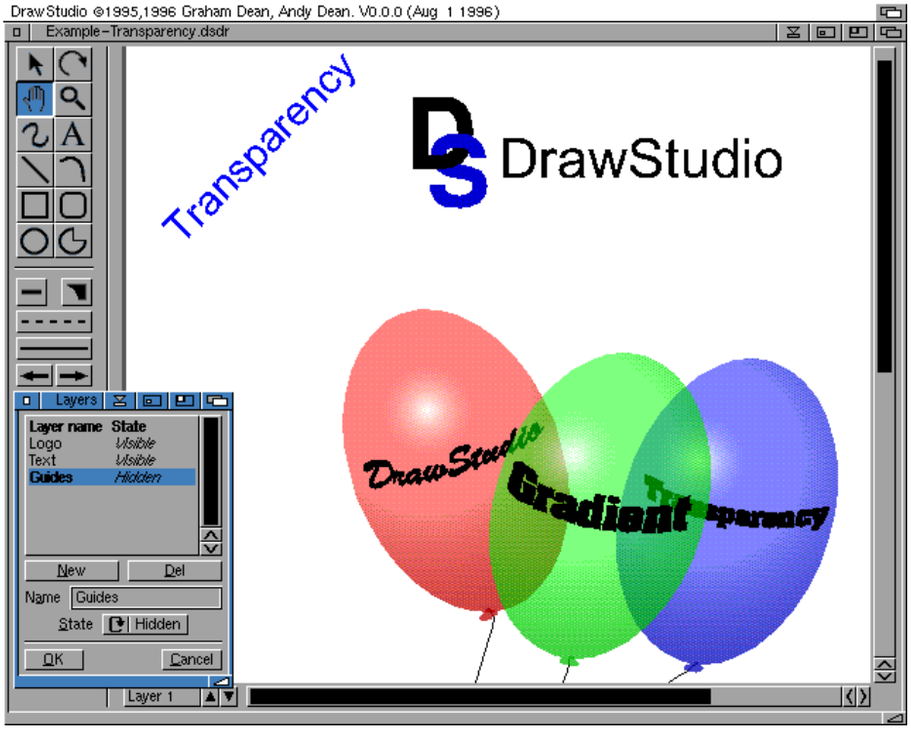
Na świecie standardem w PC są dwa programy – CorelDRAW i Adobe Illustrator. Corel jest firmą która przysypia, wolniej wprowadza innowacje, i w zasadzie już na każdym odcinku oddała pole Adobe. Choć dla mnie osobiście CorelDRAW jest programem bardziej „logicznym" i wygodniejszym w użyciu niż Ilustrator. Kiedyś był kultowy Freehand (przejęty przez Adobe tylko po to by go ubić). W tym miejscu można by rozwinąć długą opowieść o firmach niszczonych w ramach konkurencji, nierzadko oferujących bardziej innowacyjne i lepsze rozwiązania, niż zwycięzca. Próbuje obecnie przebijać się, zarówno na Apple jak i PC, Affinity Designer od Serifa, firmy z wielkimi tradycjami w produkcji oprogramowania. Niestety, nie ma miejsca na rynku na więcej. Konkurować z Adobe można jedynie oferując program działający nie w chmurze, nie wymagający subskrypcji, i z dogodnymi tanimi aktualizacjami. Technicznie Adobe raczej jest niedościgniony. Nie chodzi tylko o ilość, ale i jakość uzyskanych efektów, także zgodność ze standardami w poligrafii, po prostu Adobe jest numerem jeden.

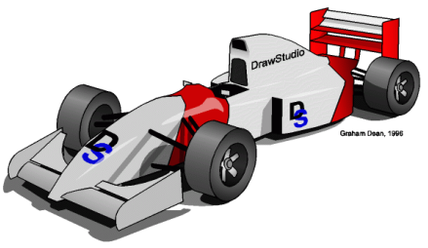
Jak w tym wszystkim prezentuje się nasz mały amigowy rynek? Chciałoby się od razu zapytać – jak bardzo jesteśmy z tyłu? Oj, bardzo.

DrawStudio

Rozwój programu zakończył się w 1997 roku. O jego możliwościach można poczytać na stronie niegdysiejszego dystrybutora, Haage und Partner Computer GmbH: www.haage-partner.de/amiga/products/ds_ft_e.htm

20 lat to kilka generacji sprzętu i oprogramowania. Zaletą DrawStudio może być dla niektórych klasyczny i schludny wygląd. Niestety co do samych możliwości to są naprawdę skromne. Stworzyć ładny napis i nałożyć na bitmapę, czy zrobić poglądowy rysunek – można. Można wycisnąć więcej – dowodem są projekty dołączone do programu, z których na kolejnej stronie pokażę dwa. Są to prace zrobione przez autorów programu.
Bolid F1 powstał na bazie grafiki wygenerowanej przez program 3D. Scyzoryk jest przykładem jak można





pomysłowością, w wykorzystaniu nielicznych narzędzi, nadrobić braki programu.

Skąd wziąć: najłatwiej z „CU Amiga magazine", a ściśle z płyty kompaktowej dołączonej do numeru na listopad 1997. Plik .iso znajduje się w bibliotece archive org: archive.org/details/cu-amiga-super-cd-rom-16

Jest to wersja lite, ale katalog ma 107 MB, i nie ma ograniczeń co do funkcjonalności narzędzi.

Po uruchomieniu może być komunikat „Cannot open screen" lub "Out of memory". Miękki reset powinien wystarczyć. Po udanym uruchomieniu trzeba ustawić większe limity pamięci w programie (nie zapomnijcie zapisać). DrawStudio działa na ściągalnym ekranie, ale w innej rozdzielczości niż blat WB, i może być konieczne poeksperymentowanie z promotorami ekranów. Ja próbowałem systemowego Screens oraz NewMode. Udało mi się uzyskać pełną rozdzielczość natywną dla swojego monitora (1920 na 1080). Trudno mi znaleźć współczesne zastosowanie dla DrawStudio. Nadaje się tylko do stawiania pierwszych kroków.

AmiFIG

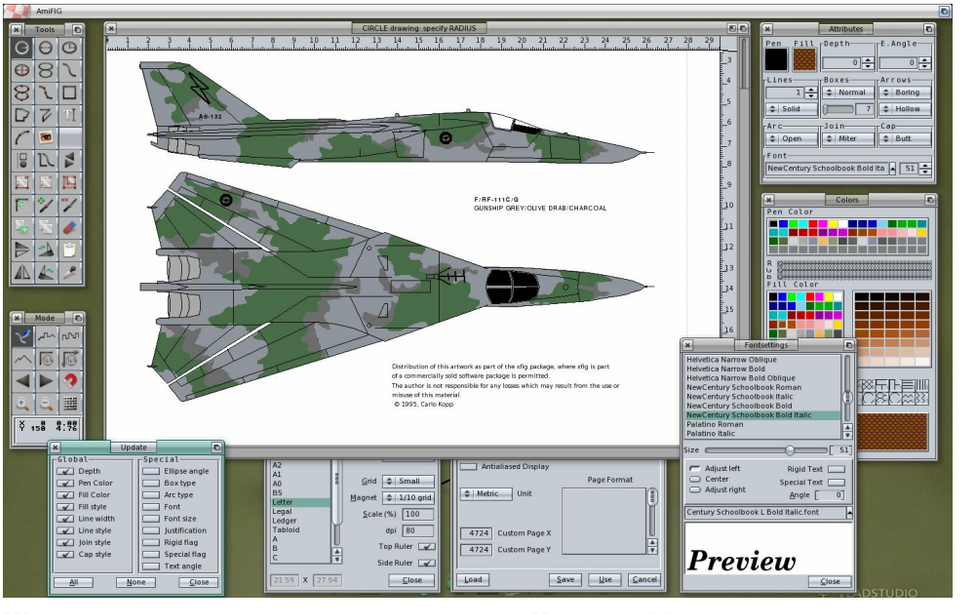
Jest to klon programu Xfig ze świata linuksa. Xfig był obecny na wielu platformach, ale współcześnie nie jest rozwijany, zakończył swój żywot w 2010 r. (ostatnia aktualizacja wyszła w 2016 r., ale to jest martwe).
Amigowa wersja Xfiga to AmiFIG, i były do niego dwa podejścia. Najpierw był rozwijany przez Andreasa Schmidta, który w pewnym momencie przeszedł na inne platformy – robi teraz komercyjny WinFIG (najnowsza wersja to 7.4), obecny na systemach Windows/macOS/Linuks.
Na szczęście AmiFIG został uwolniony, i w 2010 r. ukazała się wersja od `kas1e`, bazująca w dużej części na AROS-owych źródłach (arosową wersję rozwija Yannick Erb, a program nazywa się teraz ZuneFIG).

Więcej informacji o rozwoju AmiFIG-a dla AmigaOS 4 znajduje się na forum Amigans, w wątku „Initial steps with AmiFig on aos4": www.amigans.net/modules/xforum/viewtopic.php?forum=3&topic_id=4202&order=

Skąd wziąć: z OS 4 Depot. Może kiedyś zmienić się link, obecnie jest w kategorii Graphic/Edit a nie /Vector: os4depot.net/index.php?function=showfile&file=graphics/edit/amifig.lha.

Przy pierwszym uruchomieniu może być komunikat o braku reqtools.library. Ja pobrałem stąd: aminet.net/package/util/libs/ReqToolsUsr

Już sam rzut okiem na okno programu AmiFIG wystarczy by stwierdzić, że jest to program nowocześniejszy niż DrawStudio. Niestety, w porównaniu do współczesnych z PC, w dalszym ciągu są to skromne możliwości. Niemniej AmiFIG wydaje się na dziś tym podstawowym dla nas. Przydałaby się jakaś aktualizacja, od 2010 roku minął kawał czasu.



Widoczny na rysunku samolot pochodzi jeszcze z Xfiga, z 1995 r.

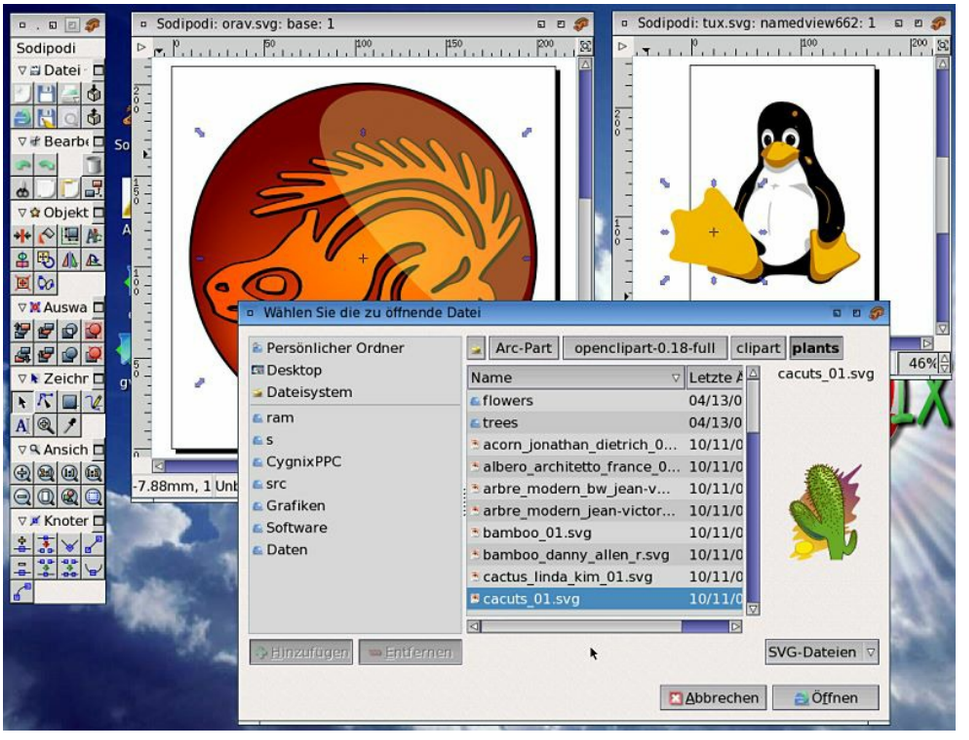
Sodipodi

To program, który do uruchomienia wymaga AmiCygnixa. Z racji tego, że sam AmiCygnix stoi od dawna w miejscu, a ja sam go nie używam, nie dotykałem się Sodipodi. Zresztą to program martwy (ostatnia wersja jest z 2009 r.), jego kontynuacją jest Inkscape (niestety nie mamy go w wersji amigowej). Wiem, że Sodipodi na współczesnych konfiguracjach działa, bo opisany jest w książce „AmigaOS 4 na co dzień" Krzysztofa Radzikowskiego. Przy okazji polecany jest, jako program pomocniczy, **Potrace**, który przekształca bitmapy na grafikę wektorową.

Oba programy znajdziemy na OS4 Depot:

os4depot.net/index.php?function=showfile&file=graphics/edit/sodipodi.lha

os4depot.net/index.php?function=showfile&file=graphics/convert/potrace.lha



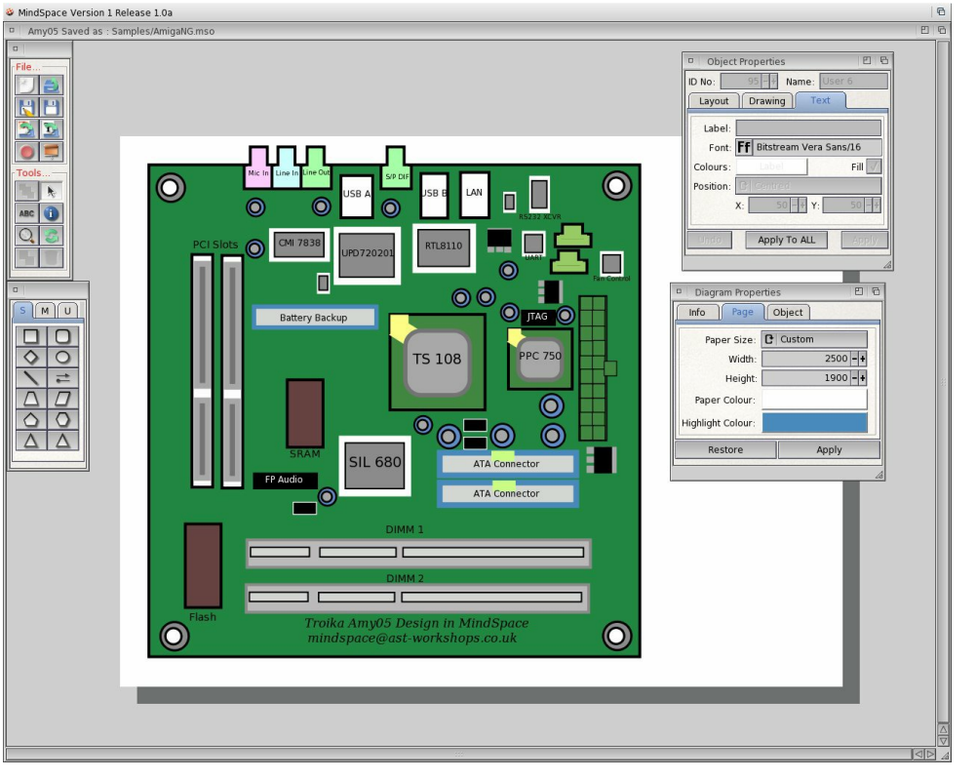
Sodipodi

MindSpace

Nie jest to typowe narzędzie do tworzenia grafiki wektorowej. Podstawowe jego przeznaczenie to tworzenie map myśli i innych schematów UML. Widoczna na ilustracji płyta główna (Amy 05 firmy Troika) pokazuje, że można wykrzesać coś więcej.

Ten program także znajdziemy „na depocie":

os4depot.net/index.php?function=showfile&file=graphics/vector/mindspace.lha

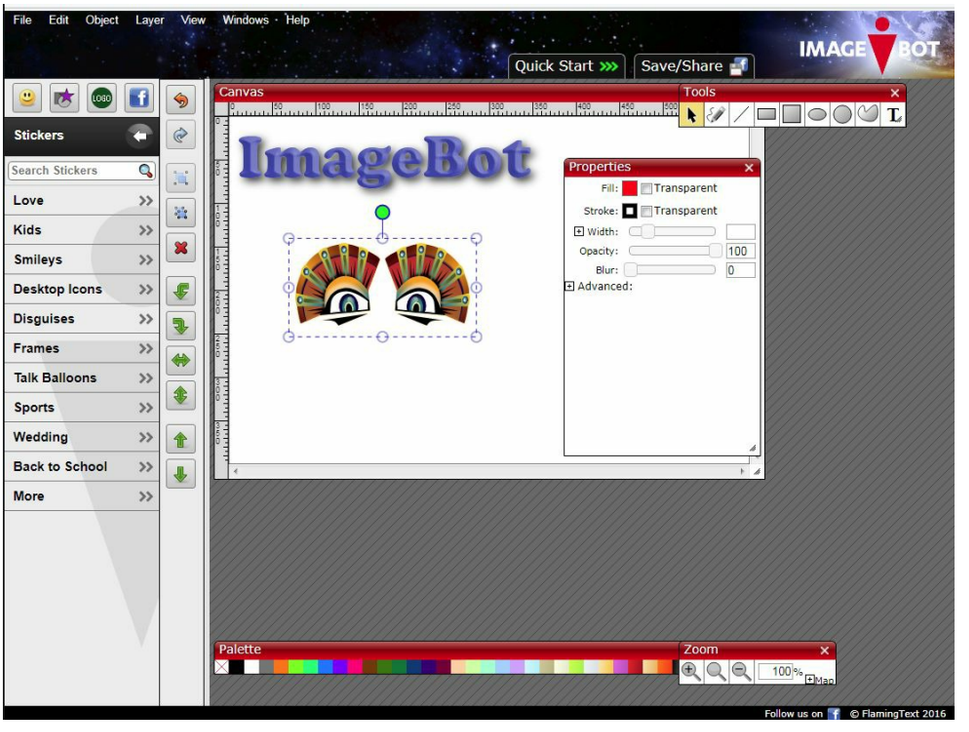


MindSpace

I to w zasadzie tyle w temacie dostępnych programów amigowych. Ale to nie oznacza koniec dostępnych opcji pod AmigaOS – czas rzucić okiem na uniwersalne rozwiązania. Są jeszcze edytory grafiki wektorowej obecne w internecie, przeznaczone do pracy online.

Edytory online

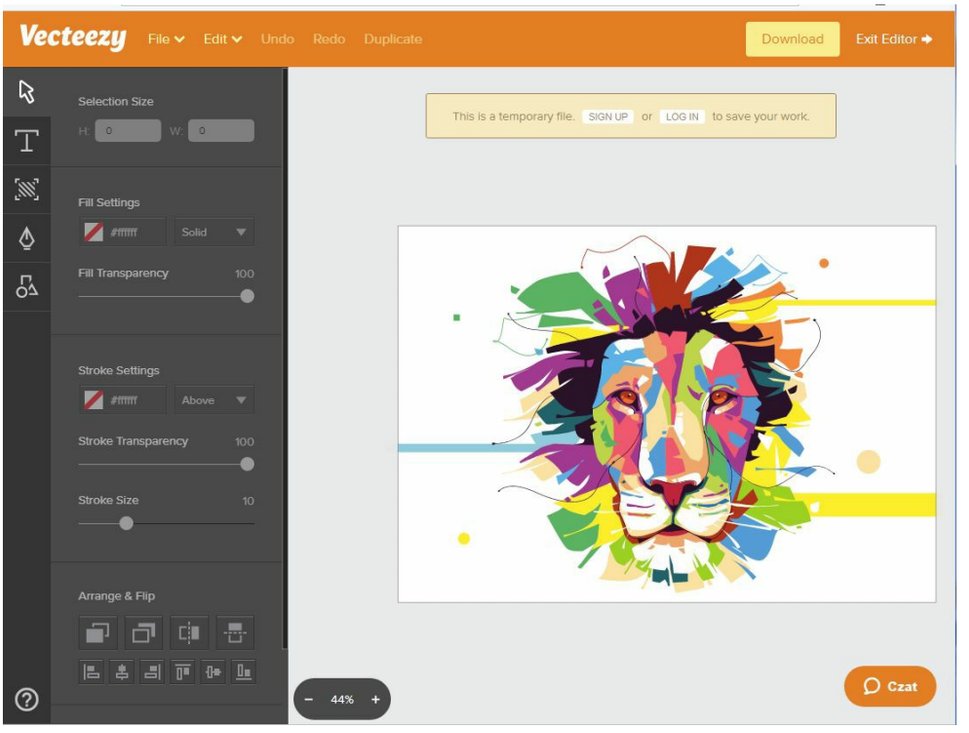
Pierwszym o jakim chcę wspomnieć jest **ImageBot**. To on niegdyś zwrócił moją uwagę na tę kategorię programów (online), gdy trafiłem na niego przypadkiem podczas lektury artykułu „Timberwolf i aplikacje webowe". Skoro istniał w 2012 roku, i jest nadal w 2017, możemy założyć że prędko nie zniknie z sieci.



Korzystanie z tej aplikacji polega głównie na umieszczaniu gotowych obiektów, pogrupowanych dla wygody w kategorie. Możemy zmieniać ich właściwości (kolor, rozmycie, przezroczystość, i inne cechy). Możemy też narysować coś od zera, ale ImageBot jest ukierunkowany raczej na złożenie jakiejś zabawnej ilustracji i podzielenia się nią na Facebooku. Stąd tyle gotowych obrazków, niemniej dzięki temu, mając na uwadze ograniczone możliwości edycji, można całkiem przyjemnie coś „ulepić".

ImageBot znajduje się pod adresem: www.flamingtext.com/imagebot/editor

Innym wartym uwagi jest **Vecteezy**: www.vecteezy.com



Żeby go uruchomić trzeba zjechać na sam dół strony, poniżej galerii, i kliknąć przycisk „Start Designing".

Po opanowaniu pewnej niezborności narzędzi, tudzież zaakceptowania jak działają (np. zalewanie płaszczyzn w miarę dodawania kolejnych węzłów) można tworzyć świetne ilustracje.
Jest to program nowszy od ImageBot. W każdym aspekcie wydaje się lepszy, począwszy od interfejsu (okna nie są „pływające"), poprzez dostępne kliparty i grafiki, skończywszy na samych możliwościach.

Widoczne poniżej okna programu obrazki to przykładowe, proponowane po uruchomieniu, sample.

Można samemu znaleźć łatwo kolejne aplikacje, wpisując w wyszukiwarkę frazy typu „vector graphics program online” czy „editor svg online”.

Rozwiązania online mają wady i zalety. Oczywistą wadą jest mało opcji, ale największa to konieczność posiadania w miarę zgodnej ze współczesnymi standardami przeglądarki internetowej. Jak wiemy ostatnio z tym jest słabo. Zaletą jest niezależność od potencjału amigowych programistów – takie aplikacje działające online będą dostępne zawsze.
Niestety JavaScript jest sporym obciążeniem, i o ile ImageBot „po namyśle” załadował się u mnie (AmigaOne 500), i mogłem w nim normalnie pracować, o tyle nowszy Vecteezy już nie. Trzeba założyć, że nowe programy wymagają komputerów klasy AmigaOne X1000 i lepszych.

Inne narzędzia

Można posiłkować się programem Hollywood Designer. Nie przedstawiłem programów które dla mnie po prostu nie warte są opisu (np. Qdiagram).

Punktem odniesienia dla mnie są programy działające pod Windows, które mam i używam – CorelDRAW X5 oraz Serif DrawPlus X8. Różnice są porażające. To już nawet nie jest zapóźnienie, to cała epoka. Nie spodziewałem się cudów, zakładałem braki funkcjonalności typu metamorfozy, ale nie aż takich różnic.

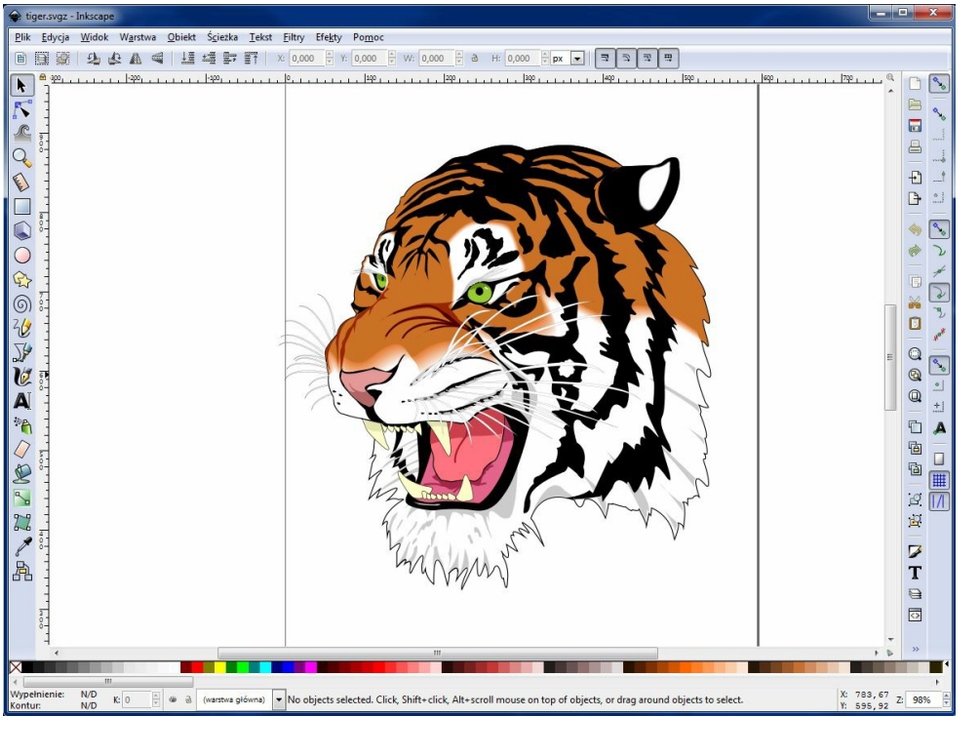
Pozostaje pogdybać o tym na co możemy liczyć w przyszłości.

Przyszłość

Moim zdaniem nowa wersja AmiFIG-a czy ZuneFIG-a nic tu nie zmieni. To byłyby półśrodki.
Inne wyjście to droga WinFiga, czyli zbudowanie aplikacji na Qt. Obadałem go pod Windows i też słabizna.

Najlepszym rozwiązaniem byłoby przeportowanie **Inkscape**.
Dostępny na wiodące platformy, za darmo i o otwartych źródłach, ładnie rozwija się. Oferuje możliwość otwarcia plików z nowszych wersji komercyjnych wiodących programów, lepsze narzędzia, jest po prostu nowocześniejszy niż AmiFIG.
Nie widzę innego wyjścia z obecnej sytuacji. Nie jest to tak nierealne, wszak mamy Blendera, który działa bez AmiCygnixa, i o którym chodzą plotki, że ma pojawić się dla AmigaOS 4 w nowszej wersji (z tych z nowszym interfejsem). Pozostaje tylko mieć nadzieję że „ktoś” to „kiedyś” zrobi. Nie da się nie zauważyć, że „ktosie” są kochane, ale to gatunek wymierający, a „kiedyś” brzmi niemal jak wyrok, tyle co „nigdy”, niemniej oceniam że jest to możliwe. Może „bounty”?

Mam przemyślenia odnośnie małej popularności Amigi NG. Zauważyłem, że częstym powodem nieużywania komputerów NG jest „odbicie się”. Ściągamy jakiś program, o którym czytaliśmy, uruchamiamy go, i... zamykamy. Nie bardzo wiemy co w nim robić, czy nawet jak używać narzędzi i opcji. Społeczność jest mała, więc jest trudno o wymianę doświadczeń.
Spróbuję to zmienić. W przyszłym numerze czasopisma mam w planie przybliżyć program AmiFIG. Myślę że nie tylko recenzje są ważne. Pokazanie jak używać programów w mojej ocenie jest niezłą zachętą.



Inkscape pod Windows 7

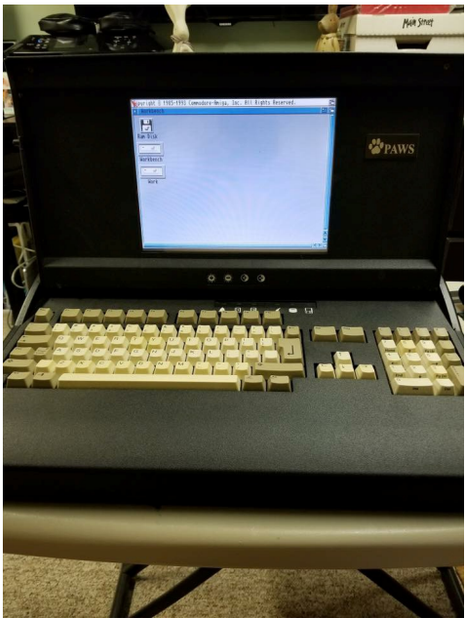
Marzenia o laptopie

truman burbank

Laptop, a właściwie jego brak, nie jest najważniejszy dla amigowców. Niemniej wiele osób chciałoby go mieć, i temat amigowego notebooka co jakiś czas powraca.

Pierwszą dla mnie konstrukcją, bo nie można nazwać tego prawdziwym laptopem, był PAWS (Portable Amiga Workstation). Pionierem był Shawn Randolph. Prace trwały prawie dwa lata, powstało kilka wariantów przeznaczonych do modeli Amiga 600, 1200 i 4000. Prawdę powiedziawszy to nawet nie były komputery – obiektem sprzedaży był zestaw obudowa plus wyświetlacz. Było to zaprojektowane w ten sposób że nie trzeba było nic lutować – wkładało się Amigę do środka. Miało to odzwierciedlenie w wyglądzie maszyny i wadze (A 600 – 11 funtów, A 1200 – 14 funtów, A 4000 – 17 funtów).
Większość zestawów była z aluminium, jednostkowe wytworzenie plastikowego prototypu było droższe.
Porty były normalnie dostępne, za wyjątkiem RGB które było wykorzystane do wyświetlacza. Użyty był zewnętrzny scandoubler Gecko. Obraz zajmował niewielka część pokrywy, oferując rozdzielczość VGA bez przeplotu.

Najbardziej dopracowany był zestaw dla Amigi 1200 – komplet zawierał CrossDos 6 Professional, oprogramowanie do przesyłania plików Link-It, oraz trackball.



Produkt był publicznie zaprezentowany w 1995, i fizycznie dostępny w sprzedaży. Trudno powiedzieć dlaczego to nie wypaliło – zapewne powodem była wysoka cena.
PAWS skończył w marny sposób – egzemplarze wypożyczone w celach demonstracyjnych sprzedawcom do składania zamówień zostały sprzedane, bądź zatrzymane na własny użytek. PAWS praktycznie nie zaistniał na rynku. W planach był normalny laptop o nazwie Puma, ale na planach skończyło się.

Kolejne lata nie przynosiły zmian w sytuacji. Powstawały różne konstrukcje „nurtu walizkowego". Nie warto o nich pisać, ponieważ były to jednostkowe egzemplarze bez możliwości zakupu, zapowiadane i w sumie niedziałające (VIScorp/QuikPak, DraCo), tudzież nierewelacyjne (np. „Susanne" zrobiona przez Simona Archera, i wiele innych). Pokażę tylko dwa egzemplarze, które zwróciły moja uwagę.

Volker Mohr zaprezentował, podczas World Of Amiga 2001 w Kolonii, przenośną Amigę 1200:

Kolorowy ekran TFT, klawiatura na podczerwień, karta turbo Blizzard 1260 50 MHz, 48 MB RAM, kontroler SCSI, dysk twardy 4 GB, karta sieciowa PCMCIA i odtwarzacz CD.

W 2009 r. anonimowa osoba wystawiła na sprzedaż na eBay ciekawą samoróbkę „Amiga 1200 Black Edition". Była to przenośna Amiga 1200, pomalowana w sposób nawiązujący do Ferrari, z ekranem TFT o przekątnej 15 cali. Wyświetlacz był obsługiwany przez podwajacz częstotliwości Indivision AGA 1200. Komputer miał wbudowane głośniki, a także odtwarzacz CD (włożony po lewej stronie). Dysk twardy 5 GB i karta turbo Blizzard 68030 50 MHz, 32 MB RAM, było dopełnieniem całości.

I w sumie to tyle co chciałem pokazać w temacie prób z oryginalną Amigą.

Volker Mohr próbował jeszcze z Pegasosem (2004/2005).
AROS w 2011 r. wszedł z dystrybucją AspireOS na laptopy Acer Aspire One 110 i 150, a potem Toshiba Satellite Pro A200.
MorphOS w 2012 wszedł na laptopy Apple PowerBook G4, w 2013 pojawiło się wsparcie części IBooków G4.

A co się działo od strony AmigaOS 4? Ano nic. W 2011 r. na AmiWest szef deweloperów Hyperionu, czyli Steven Solie, ujawnił istnienie netbooka dla AmigaOS 4. Miał być on dostępny w drugim kwartale 2012 r. w cenie 300-500 dolarów (w tym był już OS). Nigdy to się nie stało. W miarę upływu czasu po terminie powstała opinia że był to blef, mający powstrzymać amigowców od zakupu PowerBooków z MorphOS-em.
Po latach, na „Amiga 30 Germany" (2015 r.) pokazano ten niemalże „mityczny" sprzęt. Tak jak niegdyś (w 2012 r.) przypuszczano, był to LimeBook Z9. Zrobiony był tylko Kickstart, potwierdzono że prace nie będą kontynuowane. Jako powód podano niemożność dogadania się z kontrahentem co do ceny.
Nie wierzę w to. Przypuszczam że to nie cena urządzenia, a pazerność Hyperionu zabiły ten notebook. Koszt przepisania systemu i zrobienia sterowników. Jest też możliwość że zadanie okazało się za trudne.

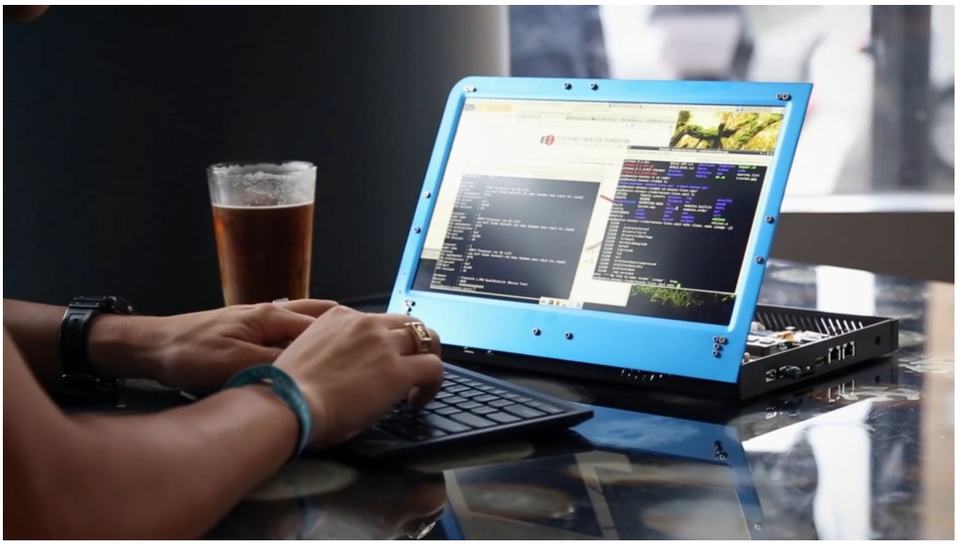
Chcę chwilę zatrzymać się nad LimeBookiem, bo w przeciwieństwie do wcześniej przedstawionych, jest to prawdziwy notebook a nie komputer mobilny, wymagający sieciowego zasilania.

Co w nim dla mnie jest niezwykłego? Moim zdaniem to ostatnia próba wejścia z PowerPC na rynek masowy. Stała za tym firma Tsinghua Tongfang. System to LimeOS, dystrybucja Linuksa w kilku wersjach: dla urządzeń wbudowanych (w tym na nasz LimeBook), mobilnej i zwykłej. Były projektory, telewizory, z netbooków tylko LimeBooki Z7 i Z9. Niestety firma nie miała szczęścia, może za wysoko się cenili, faktem jest że Z9 miał premierę jeszcze w 2008 r. i nigdy się nie upowszechnił. W 2012 była to konstrukcja przestarzała, a co dopiero teraz.
Parametry techniczne: procesor MPC5121e, wyświetlacz 9 cali, kamerka 1,3 megapiksela, 512 MB DDR 2 RAM, WiFi, dysk 32 GB SSD. Ciekawą cecha była możliwość użycia baterii „paluszków".
Istniało chyba kilka wariantów. Na Obligement był recenzowany model z 256 MB RAM, z pamięcią wewnętrzną 8 GB RAM (z tego 3,5 zajęte na start przez system). Podawana jest wydajność 1.700 MIPS-ów, oraz kilka szczegółów: jednostka Tri-Core, bateria litowo-jonowa 3.400 mAh, złącza: karta SD, mikrofon, wyjście słuchawkowe, wyjście VGA, Ethernet 10/100 Mb/s, dwa porty USB 2.0. Rozmiar: 252 x 186 x 3,5 mm, waga: 990 g.

Oznaczenie „Tri-Core" odnosi się do kilku elementów:
– procesor MPC5121e (e300c4) o częstotliwości 392 MHz
– procesor graficzny 2D/3D PowerVR MBX Lite
– jednostka audio AXIS 200 MHz: 32-bitowy AX RISC Audio Accelerator Engine

Współcześnie notebook i laptop to synonimy, ale chcąc być precyzyjnym – LimeBook to był netbook. Czyli sprzęt niewielki (mniejszy niż laptop), słaby, przeznaczony głównie do internetu. W związku z tym zachodzi pytanie: czy był sens w to wchodzić? Wiadomym jest że obecnie

minimum do znośnego korzystania z internetu to procesor powyżej 1 GHz i ponad 1 GB RAM. Procesor o wydajności 1.700 MIPS również jest za słaby, porównawczo AmigaOne 500 to 2.400 MIPS-ów, a X5000 – 12.000.
Moim zdaniem był sens. Zważywszy ceny sprzętu dla AmigaOS 4 to mógłby być hit, pełniący rolę popularyzatorską. Dziś mielibyśmy większa bazę użytkowników, i być może większy potencjał nabywczy. Bo „rynek" można rozruszać tylko tanim sprzętem.
A może temat trzeba ugryźć z innej strony – AmigaOS oferujący zamiast Workbencha interfejs w postaci nakładki do łatwej nawigacji, taki jak w LimeOS? Czyli szybki dostęp do przeglądarki, edycji dokumentów, gier.



Trochę zacząłem oddalać się od tematu. Wracając – czy pojawiło się potem coś ciekawego, godnego uwagi? Moim zdaniem tak.
Notebook „Novena". Co prawda to ARM, ale chcę przedstawić koncepcję. Każdy kupuje według potrzeb, umiejętności technicznych i zasobności portfela. Cena to od gołej płyty za 500 dolarów po zrobiony w drewnie i wykończony aluminium za 5000. Najlepsze w tym wszystkim jest to że laptop ma otwartą dokumentację:

kosagi.com//w/index.php?title=Novena_PVT_Design_Source

Jak widać na zdjęciach, „Novena" jest też dosłownie „otwarta". Takie rozwiązanie ułatwia chłodzenie (użyto pasywnego) i dostęp do podzespołów, co jest nieodzowne w takich samoróbkach.

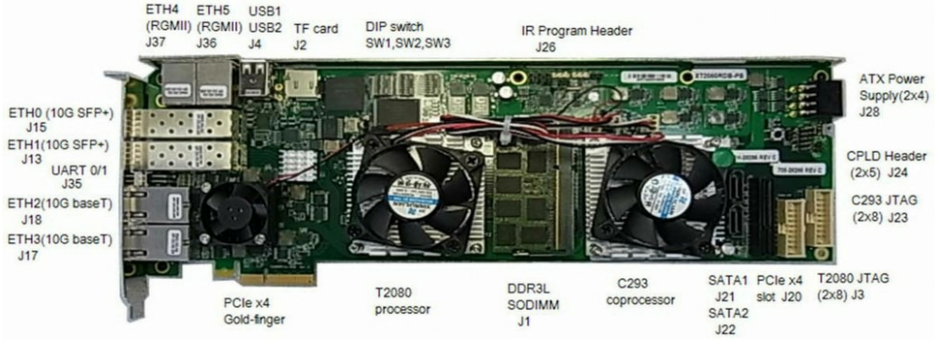
Co się tyczy limitowanej wersji w drewnie „Novena Heirloom" – robił je (bo nie było dwóch identycznych egzemplarzy) Kurt Mottweiler. Tę przepiękną robotę można podziwiać, krok po kroku, pod adresem:

mottweilerstudio.com/tag/novena-laptop/

Wchodzimy w czasy dla nas współczesne. W niusach do tego numeru podałem trzy projekty, które jednocześnie są trzema różnymi ścieżkami jakimi można dojść do amigowego laptopa.

Mniej więcej w tym samym czasie co Novena (2014 r.) pojawił się pomysł na **otwarty notebook PPC**. Inicjatorem był Roberto Innocenti. Dokładnie koncepcja została podana na Linux Day jesienią 2015 r. I od tego czasu stało to w miejscu, z braku środków i umiejętności. Dopiero tegoroczne działania przyniosły rezultaty. Po założeniu stowarzyszenia, które ma jakąś tam osobowość prawną i może organizować publiczne zbiórki pieniężne – coś „drgnęło". Kluczowe było ogłoszenie porozumienia z firmą ACube. Ponieważ dosyć obszernie podałem o co chodzi w newsie w tym numerze, w tym miejscu przeskoczę do szczegółów.

Projekt opiera się o zestaw deweloperski QorIQ T2080 Reference Design Board, oraz o Linuksa Debiana.



Jak widać na zdjęciach taki zestaw ma sporo wyprowadzonych portów. Sama płyta jest mała i dałoby się to przerobić, może nie trudniej niż Tabora. Jest dosyć szybki procesor 1,5 GHz, pamięci DDR 3, problemem jest kiepska grafika, która musi zostać rozbudowana poprzez kartę MXM. Na razie wpięty jest roboczo (na taśmie przez PCIe x4) stary Radon HD 4350.

Ponieważ bierze się za to ACube – można mieć nadzieję że AmigaOS 4 ma szansę wyjść na taki sprzęt. Zwłaszcza że ACube potrafi dostarczyć sterowniki do wszystkich komponentów płyty głównej. Byłby to fajny laptop.

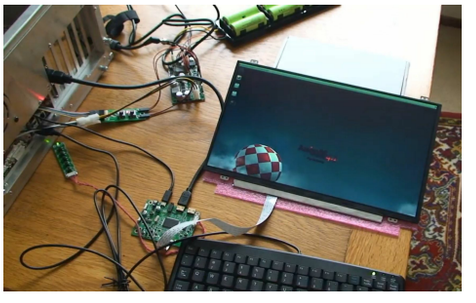
Być może trzeba będzie użyć komponentu o dokumentacji nie otwartej (NDA). Takie postawienie sprawy zniechęca do wpłat, bo to oznacza że opublikowana dokumentacja może nie być kompletna, a przecież głównym założeniem była budowa notebooka w oparciu o ideę Open Hardware.

Dlatego postęp w zbiórce odbywa się niesamowicie wolno. Można go śledzić pod podanym niżej adresem:

www.powerpc-notebook.org/campaigns/electrical-schematics-notebook-powerpc-motherboard-donation-campaign/

Kolejna opcja oparta jest o płytę Tabor. Hans de Ruiter składa laptopa, swój projekt nazwał **PORTIA** (skrót rozwija jako „ PORtable Tabor Incorporating Amigaone").
Użyto akumulatorów 93 H (w samolotach jest ograniczenie do 100 H), zużycie mocy przy pełnym obciążeniu komputera to 28 W. Nie da się ukryć, że Tabor pomimo niewielkich rozmiarów, ze względu na wyprowadzone porty, jest za wysoki na notebook. Niemniej Ruiter to jest człowiek który jak coś zacznie to zrobi. Ma firemkę, i wygląda na to, że będzie można kupić zestaw w sklepiku na jego stronie.

Na razie jest „twórczy chaos".



Jeszcze jest **A.L.I.C.E.**.
Dla mnie ten skrót jest pretensjonalny (w pełnym rozwinięciu „A Laptop Incorporating a Classic Experience"). Myślę że z czasem ludzie będą pisać po prostu „Alice". Skojarzenie z powieścią Lewisa Carrolla jest prawidłowe, użyte rozwiązanie „Rabbit Hole" ("królicza nora") jest tym co sprawia że laptop te może być przełomowy.

Były pewne perturbacje, niektórzy oceniali projekt jako martwy. Niewielkie zamieszanie wzbudziła sprzedaż egzemplarza deweloperskiego na ebayu – wystawiał Ján Zahurančík (ten od AmiKit X).

Ta konfiguracja (poszła w licytacji za 414 dolarów) to Acer Aspire V13 (dokładnie V3–371-39X4).

Jest to model z 2015 r., dosyć przeciętny – procesor Intel Core i3 4005 U (czwartej generacji) 1,7 GHz, matryca 13 cali.
Dołączono: AmiKit X, AmigaOS4, Windows, Ubuntu Mate (Linux). Rewolucyjność „króliczej nory" polega na tym że z poziomu Workbencha można wywołać programy Linuksa, czyli np. Firefoxa czy OpenOffice.

Na AmiWest 2017 pokazano nową wersję, i ta trafi do sprzedaży. To Acer Aspire E5-575 – Intel Core i3 7100 U (siódma generacja) 2.40 GHz, matryca 15 cali FullHD. Sprzedawany będzie po 650 dolarów.

Wstępne testy podają czas startu – od wciśnięcia przycisku zasilania do pokazania się ekranu wyboru systemów – 4,48 sekundy. Po wybraniu systemu – AmigaOS 4.1 ładuje się 52,78 sekundy, AmigaKit X – 22,42 sekundy.
Możliwe że będzie więcej konfiguracji, na razie z laptopów oferowany jest jeszcze Dell Latitude 5280 (mocniejszy procesor, mniejszy ekran).

Jak widać mamy trzy ścieżki:
– wykorzystanie gotowej płyty PPC o otwartej dokumentacji
– dostosowanie istniejących płyt głównych kompatybilnych z AmigaOS do obudowy laptopa
– wybranie laptopa PC i korzystanie z AmigaOS w trybie emulacji

Nie sposób nie zauważyć kilku faktów. Po pierwsze – my rozwiązujemy te same problemy co ponad 20 lat temu. Patrzę na te „komputery przenośne" przedstawione na początku artykułu, i na to co robi dzisiaj Hans de Ruiter – dla mnie to jest to samo. Po drugie – laptop z emulacją skłania (z powodu niewystarczającej wydajności) do korzystania z AmiKit (czyli OS 3.9), a nie AmigaOS 4.1.

Moja opinia na temat notebooka PPC jest taka – nie ma woli „w centrali" żeby taki model AmigaOne powstał. Jest kibicowanie (bo to nic nie kosztuje), wykorzystuje się cudze projekty w prezentacjach, w ramach promowania AmigaOS 4. Być może stoi za tym trzeźwe myślenie – głównym atutem notebooka jest jego przenośność, a główne zastosowanie to przeglądanie internetu. Ze względu na to zastosowanie, a z drugiej strony na słabą wydajność PPC w kontekście nowoczesnych stron internetowych z użyciem JavaScript – jakoś nie ma „parcia" na produkcję laptopa. Byłby to produkt po części dla idei.

Na pewno nowy amigowy notebook ucieszył by wiele osób które hobbystycznie chciałyby mieć nową Amigę, ale nie mają miejsca w domu na jeszcze jeden komputer. Prawdopodobnie przełożyłoby się to na częstsze korzystanie – nie trzeba przepinać peryferii od głównego komputera, no i można korzystać w dowolnym miejscu w mieszkaniu.

Poniżej zamieściłem zrzuty z animacji odnalezionej na YouTube. Koncepcja rozkładania notebooka przywodzi na myśl model PDA o nazwie Psion Revo (znany później jako Diamond Mako). Widać że amigowcy mają i wspomnienia, i marzenia, i wyobraźnię.

Mój opis zmagań o amigowy notebook dobiega końca. Jakie są wnioski?

Patrząc wstecz na dotychczasowe próby – szczerze wątpię w masowy produkt. Ciągle czekamy i marzymy dalej. Wcześniej czy później taki notebook może powstać, pytanie – ilu z nas dotrwa?

Hollywood #2

E.M.M.A.

ferin

Witam w drugiej części przedzierania się przez chaszcze Ostrokrzewu. Jeśli daliście radę, to gratuluję, bo z wielu powodów to była jazda przez krzaczory na oślep. Po pierwsze to był mój pierwszy artykuł w życiu; po drugie forma dziennika, jaką sobie założyłem, chyba nie do końca mi wyszła; po trzecie, pisane na szybko i nie miałem czasu wszystkiego przemyśleć i ułożyć tak jakbym chciał. Mam nadzieję, że udało mi się zachować uniwersalność kodu, tak żeby można go użyć do skompilowania binarki albo apletu na systemach innych niż przeze mnie używany AmigaOS 4.1.

Na początku sprostuję pomyłkę z poprzedniego odcinka, którą sam wychwyciłem, a mianowicie napisałem:
„Musimy pamiętać, że styl czcionki jest resetowany na domyślny po każdym użyciu funkcji SetFont czy UseFont. Czyli znowu musimy użyć SetFontColor”.
To nie jest prawda w wypadku SetFontColor. Kolor raz ustawiony będzie używany do czasu aż go nie zmienimy na inny.

Dobra, pokutę mam za sobą, teraz ogłoszenia parafialne. W związku z tym, że wpisywanie ręcznie kodu widzianego w artykule bez możliwości skopiowania go może być mozolne (jest dostępny tylko na zrzutach ekranu), pomyślałem, że udostępnię pliki tekstowe do ściągnięcia: www.czernik.me/hollywood.html

Ponadto postaram się skompilować binarki dla naszych amigowych systemów, a także dla Maca i Windy, i wrzucić je, żeby każdy mógł zassać i sprawdzić, czy i jak to działa u niego. To będzie szczególnie przydatne dla tych, którzy nie mają Hollywooda i nie mogą sami skompilować takiego programu. W pierwszej części to nie było bardzo istotne, bo wynik naszego programowania był dość prosty i nieruchomy. Inaczej będzie w tej części. Hollywood to prosty, ale całkiem mocny język i mimo swojej prostoty, pozwala na przedstawianie rzeczy na ekranie w sposób ograniczony przede wszystkim naszą wyobraźnią.

„Zaproszenie”, które było swojego rodzaju invitką na party, zostało zrobione w stu procentach, używając rozkazów/funkcji Hollwyood.
Nie ma tam wklejonych grafik czy jakichś niehollywoodowych sztuczek. Są natomiast czcionki ściągnięte z sieci. Mają one licencję pozwalającą na używanie ich do woli na użytek własny albo/i komercyjny i są zupełnie darmowe.
Podam listę, jeślibyście chcieli je u siebie zainstalować: Archistico, Elegant Lux Mager, Exo 2, Lato.

Odnośnie do przyjaznej instalacji czcionek w swoim systemie możecie poczytać pod adresem:
www.hollywood-mal.com/docs/html/hollywood/PrgG2KFonts.html

Na samym początku zajmiemy się funkcjami tworzenia obiektów tekstowych, a także wyświetlania i manipulowania nimi. Zarówno jeśli chodzi o wygląd jak i pozycję na ekranie. Bez lania wody przechodzimy do mięsa. Żeby cokolwiek wyświetlić, obrócić czy przesunąć musimy najpierw to stworzyć. W naszym tekstowym przypadku służy do tego funkcja:

CreateTextObject*(id, text$[, table])*

Działa ona jak deklaracja zmiennej w innych językach, z tą różnicą, że w naszym wypadku mamy możliwość przypisania naszej zmiennej różnych cech przekazywanych jako tablica - atrybut **[, table]**. Nasz tekst przekazujemy jako atrybut **"text$", "id"** natomiast to identyfikator, który umożliwi nam odróżnienie naszego tekstu. Jeśli pominiemy opis cech przekazywanych poprzez [, table], to nasz tekst będzie wyświetlony z domyślnymi lub wcześniej w skrypcie ustawionymi atrybutami dla czcionki.

Mała dygresja odnośnie do dziwnego zapisu składni funkcji – wygląda tak, jakby przecinek był zamieniony miejscami z nawiasem kwadratowym – **[,table]**. Nie

wiem, dlaczego ona tak wygląda, ale tak jest w oficjalnej dokumentacji, więc tego się będę trzymał.

Wracając do funkcji. Ona jedynie tworzy obiekt tekstowy, niczego nie wyświetla, ani nie wypisuje. Poza tym, co przypadkiem zdarzyło mi się odkryć, możemy stworzyć jeden obiekt i wyświetlić go wielokrotnie. Nie wiem czy takie wykorzystanie funkcji idzie w parze z wytycznymi poprawnego tworzenia skryptów w Hollywood, ale w tym wypadku działa i nic się nie kaszani.

W argumencie table, możemy określić:
Align= wyrównanie
#Left – do lewej (wartość domyślna)
#Right – do prawej
#Center – po środku
#Justified – równomierne rozłożenie – **ten argument wymaga Hollywood w wersji 7.0!**

WordWrap= szerokość w pikselach, na przykład 500

Encoding= system kodowania znaków

Color= kolor tekstu, musi być podany jako wartość **ARGB**, czyli w postaci **$AARRGGBB**

Tu muszę się podzielić następnym odkryciem, bo nieopisanym w dokumentacji; otóż jeśli określimy WordWrap, to tekst podany w argumencie tekst$ musi znajdować się na jednej linii w skrypcie, w innym wypadku interpreter wyświetli nam błąd: „Unterminated string!”; w wypadku krótkich tekstów to nie kłopot, ale w wypadku dłuższych może to się okazać problematyczne. Pewnie da się przekazać tekst jaki ma stworzyć nasza funkcja jako wskaźnik do pliku, z którego sama odczyta treść, ale my jeszcze tego nie potrafimy. Natomiast jeśli nie określimy WordWrap to tekst wyjedzie nam poza ekran jeśli będzie zbyt długi.

Zanim zaczniemy odnosić powyższe informacje do konkretnego przypadku naszej invitki, muszę jeszcze wspomnieć o tym, w jaki sposób Hollywood wyświetla obiekty na ekranie. Odnosi się to do wszystkich obiektów, jakie są dostępne w tym języku programowania.

Warstwowość

Jako że Hollywood to język skryptowy, to jest odczytywany linia po linii od góry do dołu, Amerykę w konserwach...
Przez co wszystko, co wyświetlimy, będzie „układane” jedno na drugim, czyli jeśli na początku wrzucimy sobie słowo „Cześć” na połowę strony, to nasze powitanie będzie w tle tego, co wyświetlimy później. Warto to wziąć pod uwagę, bo dopóki nie nauczymy się tworzyć warstw i nimi manipulować, nie mamy żadnej możliwości zmiany kolejności ich wyświetlania na ekranie.
Warstwami zajmiemy się, gdy pojawi się taka potrzeba, sama biblioteka rozkazów warstw jest tak duża, że mogłaby wypełnić kilka artykułów. A nie jest moim celem „przepisywanie” dokumentacji, więc najpewniej opiszę tylko te funkcję potrzebne do naszych celów.

Podsumowując: Hollywood będzie wyświetlał nasze obiekty na ekranie tak, jakbyśmy my kładli kolejne kartki papieru lub przeźrocza jedne na drugie.
Mając to na uwadze, możemy zacząć tworzyć nasze zaproszenie. Na początku mamy znajome funkcje, których po raz wtóry nie będę tu opisywał.
Ogólnie tylko pisząc, mamy tam określenie wymaganej wersji Hollywood, wielkości ekranu, koloru tła, bezramkowość okna i polecenie załadowania pięciu krojów czcionek do późniejszego użycia. *(Zrzut 1)*.

```
@VERSION 7,0                                                    ; wymagana wersja Hollywood
@DISPLAY{
                Width=1000,                                     ; szerokość okna
                Height=800,                                     ; wysokość okna
                Borderless=True,                                ; okno bez ramki = tak
                Color=$F0F0F0                                   ; kolor tła
        }

@FONT 1, "Elegant Lux Mager", 220, {Engine=#FONTENGINE_INBUILT, Link=False }        ; czcionka dla obróconego "Cześć"
@FONT 2, "Exo 2 Italic", 48, {Engine=#FONTENGINE_INBUILT, Link=False}               ; czcionka dla tekstu Zaproszenia
@FONT 3, "Exo 2 Bold", 700, {Engine=#FONTENGINE_INBUILT, Link=False}                ; czcionka dla "#2"
@FONT 4, "Archistico Bold", 170, {Engine=#FONTENGINE_INBUILT, Link=False}           ; czcionka dla obróconych ">>>"  "ptaszków"
@FONT 5, "Lato Thin", 32, {Engine= #FONTENGINE_NATIVE, Link=False}                  ; czcionka dla wytłumaczenia z "*"
```

Zrzut 1

Następnie przy pomocy funkcji **UseFont** ustawiamy aktualną czcionkę na **Font "3"** czyli **Exo 2 Bold** o wielkości **700** pikseli.
Zmieniamy kolor – **SetFontColor,** i włączamy wygładzanie – **SetFontStyle(#ANTIALIAS)**. Gdy mamy już określone, jaką czcionką chcemy pisać, to możemy stworzyć obiekt tekstowy, który będzie nią pisany; w drugą stronę nie zadziała. Mam na myśli, że najpierw musimy określić czcionkę i jej cechy, a dopiero potem tworzyć obiekt tekstowy. Teraz wyświetlmy nasz obiekt na ekranie za pomocą:

DisplayTextObject*(id,x,y)*

Funkcja ta przyjmuje trzy argumenty. Pierwszy to identyfikator tekstu który chcemy wyświetlić, a x i y jak łatwo się domyślić to współrzędne na ekranie. Układ współrzędnych w Hollywood ma swój początek, czyli 0,0, w lewym górnym rogu ekranu. Przy tej okazji muszę wspomnieć o tym, że każdy obiekt ma punkt, który jest jego „uchwytem"; w Hollywood nosi on nazwę **Anchor** – kotwica, jest on zarazem osią obrotu.
Domyślnie jest on ustawiony na górny lewy róg niewidocznej ramki obramującej obiekt, która może przybierać kształt prostokąta.
Punkt uchwycenia można zmieniać, niestety nie można go przesunąć poza obręb obiektu, na przykład w punkt -1, -1, ale o tym przy innej okazji.

DisplayTextObject(3, 60, -50)

Liczba 3 w nawiasie to identyfikator naszego tekstu, który nadaliśmy mu przy tworzeniu go funkcją CreateTextObject (3, "#2"), teraz przesuwamy nasz napis 60 pikseli w prawo i 50 w dół.
Czyli do kodu widocznego powyżej dopisujemy kod jak na *(Zrzut 2)* i otrzymujemy rezultat jak na *(Zrzut 2.1)*.
(Na zrzutach pokazuję dopisywany kod, cały listing – pod koniec artykułu, na str.19).

Pokrótce tylko opiszę funkcję bardzo podobną do poprzedniej. Użycie jej w naszym skrypcie nie ma sensu, bo wynik jest nieruchomy, a funkcja ta określa efekt specjalny, z jakim na ekranie pojawiają się obiekty tekstowe.

DisplayTextObjectFX*(id, x, y[, table])*

Jak widać, jedyna różnica to dodatkowy atrybut **[, table]**. W bardzo dużym skrócie – jest to tablica opisująca, jaki efekt ma być użyty i z jaką prędkością.
Przejdźmy teraz do następnego elementu naszej invitki, a są to trzy rzędy obróconych pod kątem zielonych znaków „większy niż".
Obracanie obiektów tekstowych odbywa się przy pomocy funkcji:

RotateTextObject*(id, angle)*

Składnia jest prosta, **id** to identyfikator obiektu tekstowego który chcemy obrócić, a **angle** to kąt o jaki obracamy. Wartość dodatnia obraca odwrotnie do wskazówek zegara, a ujemna zgodnie z nimi. Ważna uwaga odnośnie obracania czcionek: Hollywood nie potrafi jeszcze

Zrzut 2.1

```
; <<<<<<<<<<<<<<<<<<<<<<          Wielkie "#2" w tle     >>>>>>>>>>>>>>>>>>>>

UseFont(3)
SetFontColor($E0E0E0)
SetFontStyle(#ANTIALIAS)
CreateTextObject(3, "#2")
DisplayTextObject(3, 60, -50)

WaitLeftMouse()
End()
```

Zrzut 2

obracać czcionek wektorowych i przez to zmienia je na obiekt graficzny, dlatego musimy ustalić wszystkie cechy jakie chcemy nadać naszemu tekstowi przed obrotem, (ma się to zmienić, niestety w niekreślonej przyszłości). W wypadku czcionek wektorowych wiąże się to z utratą jakości, w przypadku bitmapowych oczywiście nie ma to znaczenia, bo już są pikselowe przed obrotem.

To do dzieła. Używamy czcionki z id 4, zmieniamy kolor na jakiś zielony – $D0E0D0, i tworzymy obiekt tekstowy „**\t \t \t >>>"**
Te trzy „**t**" to trzy znaki tabulacji, które chcemy wstawić przed naszymi „ptaszkami" w celu przesunięcia osi obrotu poza obręb widocznej części obiektu tekstowego. Moglibyśmy użyć atrybutu AnchorX i Y, ale tak jak wspomniałem wcześniej, nie da się przesunąć punktu uchwycenia dalej niż sięga sam obiekt, a w tym wypadku to by nie wystarczyło. Stąd taka bandycka metoda wstawienia kilku białych znaków – czyli matki, potrzeby i wynalazki, czy jak tam to szło.
Pierwszy rząd „ptaszków" obracamy o 45 stopni, czyli z grubsza na dół i w prawo – **RotateTextObject(4, -45)**, następnie musimy go wyświetlić, co czynimy przez **DisplayTextObject(4, #CENTER+250, #CENTER+180)**
Wartości dobierałem metodą prób i błędów, zarówno jeśli chodzi o obrót jak i pozycję na ekranie. Napiszę tylko co to ten #CENTER+250 i 180, jest to pozycja dla wyświetlenia naszego obiektu tekstowego o id 4.
#CENTER to środek naszego okienka z invitką, a wartości 250 i 180 pikseli dodałem do automatycznie wyliczonego środka okienka. Zamiast wyliczać co do piksela wysokość i szerokość, dzielić na pół i dodawać powyższe wartości, zastosowałem taką metodę, stosowanie stałych może ułatwić tworzenie programów niezależnych od rozdzielczości ekranów, na których są wyświetlane. Oczywiście trzeba by wtedy użyć wartości względnych w stosunku do wielkości naszego okna, czcionek i tak dalej, ale to zupełnie inna bajka.
O.K., w tym miejscu mamy już jeden rząd ptaszków na ekranie, teraz za pomocą RotateTextObject obracamy go jeszcze raz i wyświetlamy ponownie, i potem jeszcze raz to samo, oczywiście z innymi wartościami. Kod i wynik poniżej na zrzutach *(Zrzut 3)* i *(Zrzut 3.1)*.

Zwróćcie uwagę na dwie rzeczy. Pierwsza to fakt że obiekt tekstowy został stworzony raz, a wyświetlony trzykrotnie. Druga to to, że obrót następuje zawsze od pozycji wyjściowej. W naszym wypadku stworzyliśmy jeden obiekt, obracaliśmy go kolejno o wymaganą liczbę stopni, czyli nie jest tak, że:
tworzymy –> obracamy –> wyświetlamy –> obracamy dalej, tylko:
tworzymy –> obracamy –> wyświetlamy –> **obracamy od zera** –> znowu wyświetlamy...

Zrzut 3.1

```
; <<<<<<<<<<<<<<<<<<<<<<          Zielone "ptaszki"      >>>>>>>>>>>>>>>>>>>>

UseFont(4)
SetFontColor($D0E0D0)
CreateTextObject(4, "\t \t \t >>>")
RotateTextObject(4, -45)
DisplayTextObject(4, #CENTER+250, #CENTER+180)
RotateTextObject(4, -65)
DisplayTextObject(4, #CENTER+170, #CENTER+180)
RotateTextObject(4, -90)
DisplayTextObject(4, #CENTER+90, #CENTER+180)


WaitLeftMouse()
End()
```

Zrzut 3

Następna część naszego skryptu technicznie nie różni się od poprzedniej. Wybieramy czcionkę, ustawiamy kolor, włączamy wygładzanie, tworzymy obiekt tekstowy „CZEŚĆ”, nadajemy mu identyfikator 1, dodając równomierne rozłożenie (#JUSTIFIED) na odcinku pięciuset pikseli (WordWrap=500).
Następnie obracamy o 90 stopni i wyświetlamy po prawej stronie na środku, używając stałych #RIGHT-20, #CENTER. Kod i efekt na zrzutach *(Zrzut 4)* i *(Zrzut 4.1)*.

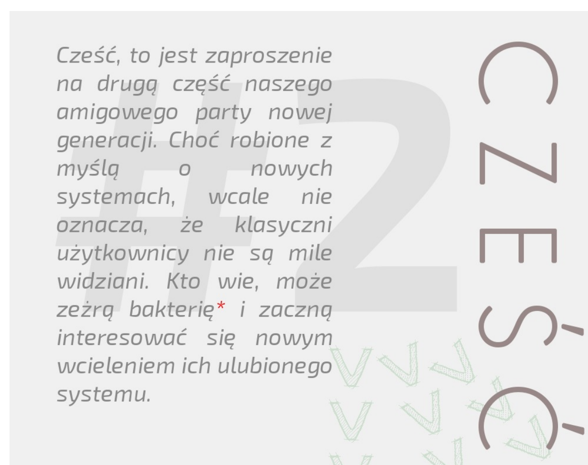
Przyszedł czas na główny tekst zaproszenia, czyli szpalta literek po lewej stronie na pierwszym planie. Tu nie będę się rozpisywał, bo poza jedną rzeczą nie ma dla nas nic nowego. Kod i zrzut poniżej, podpisane jako *(Zrzut 5)* i *(Zrzut 5.1)*.

Zrzut 4.1

```
; <<<<<<<<<<<<<<<<<<<<<          Obrócone "Cześć"          >>>>>>>>>>>>>>>>>>>

UseFont(1)
SetFontColor($998888)
SetFontStyle(#ANTIALIAS)
CreateTextObject(1, "C Z E Ś Ć", {Align=#JUSTIFIED, WordWrap=500 })
RotateTextObject(1, -90)
DisplayTextObject(1, #RIGHT-20, #CENTER)
```

Zrzut 4

Jedyną rzeczą, którą już opisywałem w poprzedniej części, jest formatowanie tekstu „w locie”. W naszym wypadku zmieniłem kolor gwiazdki odniesienia na czerwony tagiem formatującym
[COLOR=#RED]*[/COLOR]
w tekście przekazywanym jako argument text$, funkcji CreateTextObject. Poza tym, tak jak już wspominałem wyżej, cały tekst musi się znajdować na jednej linii skryptu.



Zrzut 5.1

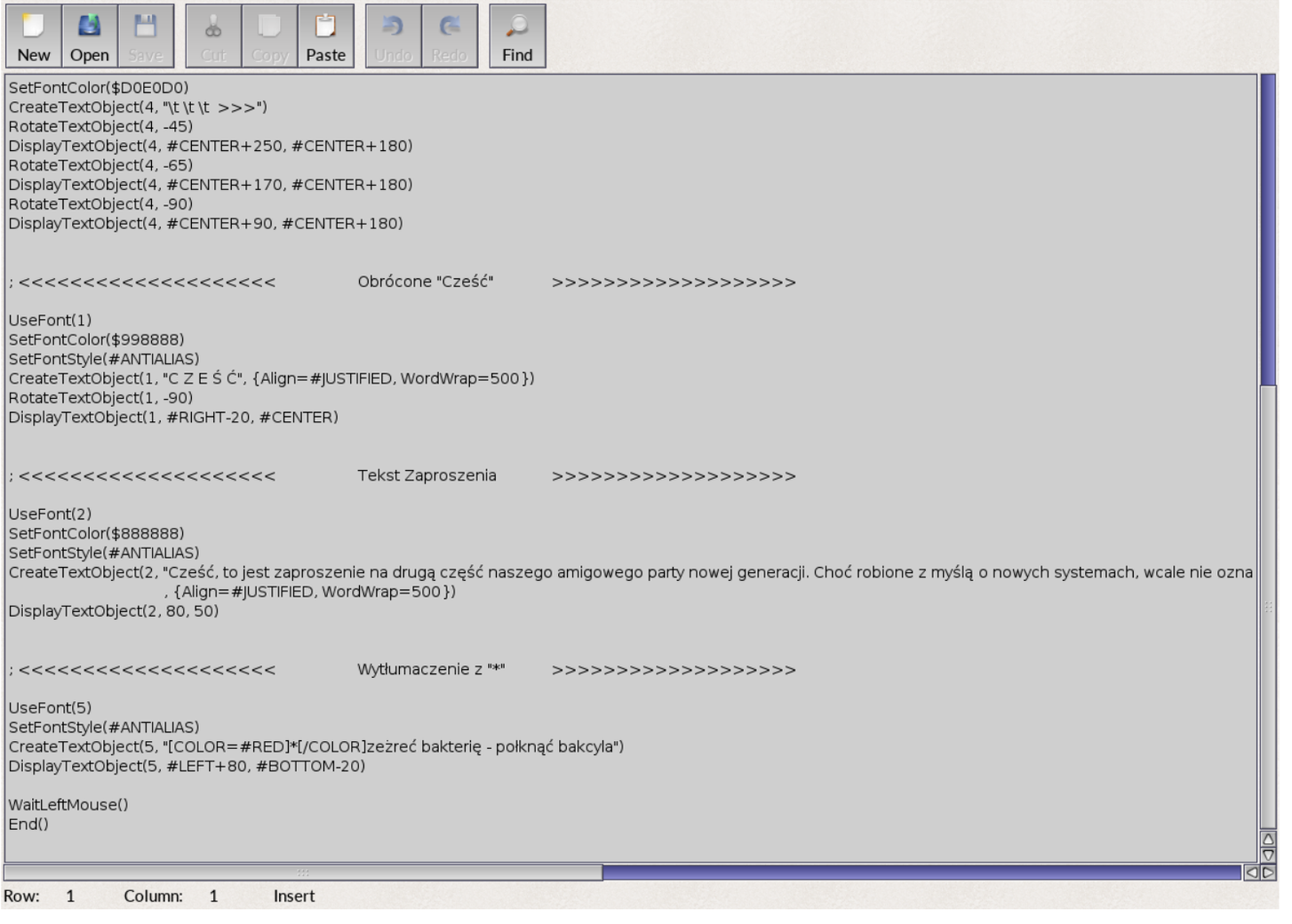
```
; <<<<<<<<<<<<<<<<<<<<<          Tekst Zaproszenia          >>>>>>>>>>>>>>>>>>>

UseFont(2)
SetFontColor($888888)
SetFontStyle(#ANTIALIAS)
CreateTextObject(2, "Cześć, to jest zaproszenie na drugą część naszego amigowego party nowej generacji. Choć robione z myślą o nowych systemach, wcale nie ozr
                        , {Align=#JUSTIFIED, WordWrap=500 })
DisplayTextObject(2, 80, 50)


WaitLeftMouse()
End()
```

Zrzut 5

W końcu ostatni element zaproszenia, którym jest tekst wyjaśniający po gwiazdce na samym dole okienka. Tu nie ma już zupełnie nic nowego dla nas, więc rozkminienie pozostawiam Wam. Kusiło mnie natomiast żeby zastosować inną funkcję do wyświetlenia tego tekstu, a mianowicie TextOut, ale ta niepozornie wyglądająca funkcja to gigant jeśli chodzi o możliwości; ma oczywiście swoje ograniczenia, ale w zależności od potrzeb może zastąpić CreateTextObject i DisplayTextObject jednocześnie, obiec je dwa razy w koło, zdrzemnąć się, a tamte dwie i tak wciąż będą z tyłu. Dlatego opiszę ją na końcu.

Zrzuty z ostatnią częścią invitki: *(Zrzut 6)* i *(Zrzut 6.1)*, czyli cały kod zawarty w lekcji i efekt końcowy.

Zrzut 6.1



Zrzut 6

Żeby dopełnić opis funkcji manipulujących tekstem opiszę kilka funkcji, które mogą nam się przydać w przyszłości.

[id] = **CopyTextObject***(source, dest)*

Funkcja ta kopiuje obiekt tekstowy „source”, który musimy podać jako identyfikator do źródłowego obiektu tekstowego, masło maślane, do innego obiektu tekstowego, który musimy podać jako identyfikator w miejsce „dest”. Możemy zdać się na Hollywood, i nie podawać identyfikatora docelowego, i zamiast niego użyć **NIL**; wtedy zostanie automatycznie przydzielony identyfikator, i zwrócony jako wynik działania funkcji do [id], które to musimy określić.

Na przykład

```
CreateTextObject(1, "Tekst do sklonowania")
DisplayTextObject(1, 20, -10)
sklonowany = CopyTextObject(1, NIL)
DisplayTextObject(sklonowany, 20, -100)
```

stworzy nam obiekt tekstowy „Tekst do sklonowania” wyświetli go 20 pikseli od lewej strony i -10 pikseli od góry, skopiuje go i wyświetli 20 pikseli od lewej strony i -100 pikseli od góry. Ograniczeniem tej funkcji jest to, że klon nie poddaje się żadnym próbom zmiany wyglądu – będzie wyglądał tak jak oryginał. Można oczywiście klona obracać, przesuwać itp. Szczerze pisząc, nie wiem jakie może być zastosowanie tej funkcji, wydaje się dawać mało korzyści w stosunku do narzucanych ograniczeń, ale jestem przekonany, że jest jakiś dobry powód jej istnienia, tylko jeszcze o nim nie wiem ;).

Bez kombinacji z autoprzydzielaniem identyfikatorów to samo co wyżej będzie wyglądać tak:

```
CreateTextObject(1, "Tekst do sklonowania")
DisplayTextObject(1, 20, -10)
CopyTextObject(1, 2)
DisplayTextObject(2, 20, -100)
```

FreeTextObject*(id)*

Ta funkcja uwalnia nam obiekt tekstowy z pamięci, dzięki czemu możemy ją oszczędzić na coś bardziej akurat potrzebnego ;D. Przykład:

```
CreateTextObject(1, "Tekst do sklonowania")
FreeTextObject(1)
```

MoveTextObject*(id, xa, ya, xb, yb[, table])*

Przesuwa obiekt tekstowy o identyfikatorze „id” z punktu „a” do punktu „b”; **„a”** i **„b”** to koordynaty w pikselach. Poza tym opcjonalnie w tablicy możemy określić z jaką szybkością ma następować przesunięcie, czy chcemy zastosować efekt specjalny towarzyszący zmianie pozycji, i w końcu czy przesunięcie ma być asynchroniczne. Nie będę tego zgłębiał teraz, ale jeśli was to ciekawi to możecie poczytać o tym pod adresem:

www.hollywood-mal.com/docs/html/hollywood/MoveBrush.html

ScaleTextObject*(id, width, height)*

Funkcja ta skaluje obiekt tekstowy określony identyfikatorem id do pożądanych rozmiarów. Jeśli nasz tekst jest pisany czcionką wektorową to nie nastąpi utrata jakości. Jako **width** albo **height** możemy podać stałą wbudowaną **#KEEPASPRAT** (od Keep Aspect Ratio – zachowaj proporcje) wtedy rozmiar zostanie obliczony automagicznie. Width i height mogą zostać podane jako liczby albo procenty. Przykłady:

ScaleTextObject(1, 200%, #KEEPASPRAT) – powiększy tekst wszerz dwukrotnie i automatycznie dopasuje wysokość zachowując proporcje.
ScaleTextObject(1, #KEEPASPRAT, 500) – zmieni wysokość obiektu tekstowego do 500 pikseli i dopasuje automatycznie szerokość.

TextOut*(x, y, text$[, table])*

Funkcja ta wypisuje tekst przekazany w argumencie "text$" w pozycji określonej przez **x** i **y**, podobnie jak w przypadku DisplayTextObject współrzędne możemy podawać jako liczby albo stałe wbudowane. a tekst może być formatowany w locie poprzez użycie tagów formatujących jak [I][/I], [COLOR][/COLOR] itd. W argumencie **[, table]** przekazujemy atrybuty podobnie jak w CreateTextObject, ale oprócz nich możemy jeszcze użyć **StandardDrawTags**.

Poczytacie o nich więcej tu: www.hollywood-mal.com/docs/html/hollywood/StandardDrawTags.html

Jak zaraz zobaczycie, funkcja ta łączy w sobie możliwości Display/Create-TextObject, i dodaje jeszcze kolejne. Należy jednak pamiętać że część opcji wymaga włączenia warstw.

Kod i efekt na zrzutach *(Zrzut 7)* i *(Zrzut 7.1)*. Jak widać są całkiem spore możliwości.

I tym kończymy drugi odcinek. Zapraszam na następny, w którym postaram się w twórczy sposób wykorzystać to, co już umiemy i dodać nowe rzeczy.

Zrzut 7.1

New Open Save Cut Copy Paste Undo Redo Find

```
@VERSION 7,0                                                    ; wymagana wersja Hollywood
@DISPLAY{
                Width=1000,                                     ; szerokość okna
                Height=800,                                     ; wysokość okna
                Borderless=True,                                ; okno bez ramki = tak
                Color=$F0F0F0,                                  ; kolor tła
                DragRegion =                                    ; umożliwia przeciąganie okna
                      {
                           {Type = #BOX, X=0, Y=0, Width=1000, Height=30}
                      },
                Layers=True                                     ; włącza warstwy, musimy je włączyć aby
                                                                ;  zademonstrować rzucanie cienia przez tekst
          }


@FONT 2, "Exo 2 Italic", 100, {Engine=#FONTENGINE_INBUILT,  Link=False}

UseFont(2)
SetFontColor($E0E0E0)                           ; Kolor goólny dla czcionki, który w ramach demonstracji zostanie
                                                ; przekolorowany, raz "w locie" i drugi raz za pomocą opcjonalnego
                                                ; tagu Standard Draw Tag
SetFontStyle(#ANTIALIAS)
TextOut(450, 450, "Nasz full wypas Text[COLOR=#GREEN]Out[/COLOR]owy napis, juhuuu!",
                {
                WordWrap=600,                   ; Zawijamy tekst
                Color=$11FF996F,                ; Kolor i stopień przeźroczysości tekstu
                AnchorX=0.5, AnchorY=0.5,       ; Zmieniamy punkt obrotu na środek
                Rotate=-45,                     ; Obracamy o 45 stopni

                Shadow=True,                    ; Rzucamy cień - wymaga włączenia warstw
                ShadowColor=$AA333333,          ; Kolor i stopień przeźroczystości cienia
                ShadowRadius=2,                 ; Wygładzanie cienia
                ShadowSize=20,                  ; Tak macie rację, to jest odległość cienia od tekstu(!?)
                ShadowDir=#SHDWSOUTHWEST,       ; Kierunek rzucanego cienia - w tym wypadku - na południowyzachód

                Border=True,                    ; Obrys = tak
                BorderSize=2                    ; Grubość obrysu dwa piksele
                }
          )
```

Row: 1 Column: 1 Insert

Zrzut 7

MorphOS i MacOS na jednym dysku

trOLLO

I. Drobny wstęp.

MorphOS jest, jak na nasze amigowe zainteresowania, całkiem ciekawym systemem, jednakże czasami przychodzi potrzeba uruchomienia jakiegoś programu/gry, którego nie ma w wersji dla MorphOS-a. W takiej sytuacji z pomocą może przyjść system MacOS X.

Poniżej postaram się pokazać, jak w miarę prosty sposób zainstalować na jednym dysku MorphOS 3.9 jak i MacOS X 10.5.8.

Sposób ten został sprawdzony na następujących komputerach:
– Power Mac G4 MDD – G4 1.25 GHz, 2 GB RAM, HD 250 GB SATA
– PowerBook G4 – G4 1.5 GHz, 1 GB RAM, SSD 120 GB mSATA
– Power Mac G5 – Dual G5 2.0 GHz, 4 GB RAM, HD 1 TB SATA
Potrzebne nam będą płytki instalacyjne z systemem MorphOS oraz MacOS X (w opisywanym przykładzie zastosowałem płytkę z MorphOS 3.9 i MacOS 10.5.1).

II. Zabawę czas zacząć.

Poradnik ten można zastosować dla dwóch różnych sytuacji:

Sytuacja pierwsza
Posiadamy czysty dysk i chcemy instalować MorphOS oraz MacOS.

Sytuacja druga
Posiadamy dysk z zainstalowanym MorphOS-em i chcemy „dołożyć" MacOS X.

Tak naprawdę cały proces różni się tylko tym, że w przypadku pierwszym, musimy zainstalować zarówno MorphOS-a, jak i MacOS, a w przypadku drugim, tylko MacOS.

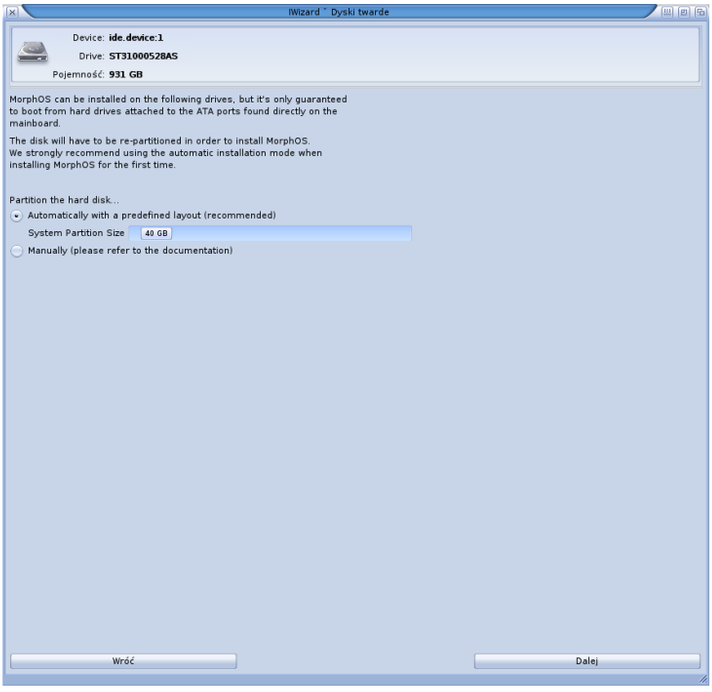
CZYSTY DYSK

Cały proces jest naprawdę bardzo prosty, więc zaczynamy! :-).
Instalujemy MorphOS 3.9, w momencie wyboru wielkości dysku systemowego (Rysunek 1.) wybieramy opcję *Automatically with…*, ustawiamy sobie wielkość partycji systemowej (u mnie 40 GB) i klikamy przycisk *Dalej*.

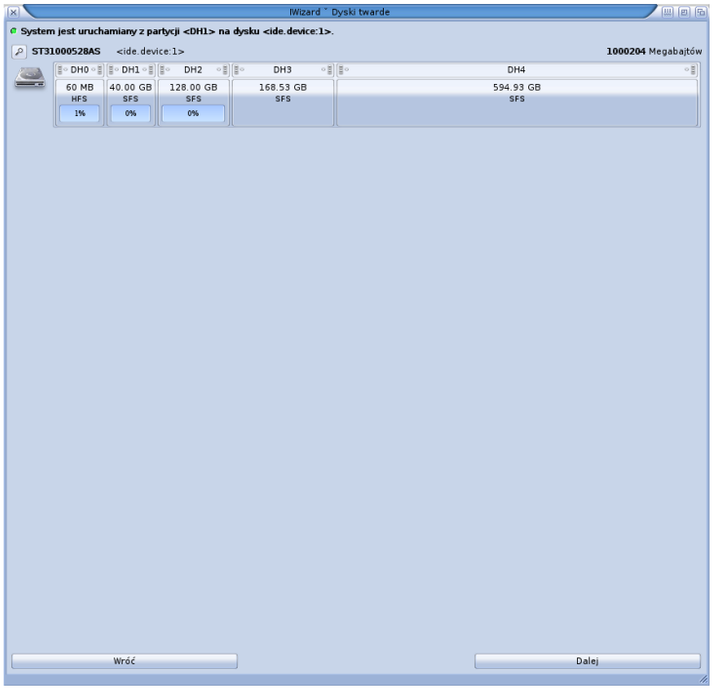
Następnie pojawi się okno z podziałem dysku na partycje. Tutaj możemy jeszcze ewentualnie pozmieniać ustawienia tak, jak nam pasuje (Rysunek 2).

W naszym przypadku:
DH0 – to partycja, z której nasz Power Mac startuje i wczytuje MorphOS-a (tzw. Partycja BOOT)
DH1 – to partycja systemowa (tutaj zainstalujemy system MorphOS 3.9)
DH2 – to partycja na wszystkie inne rzeczy (gry, muzyka, zdjęcia… cokolwiek mam pod ręką: –)), ta partycja będzie się nazywać *Work*
DH3 – tę partycję chcę wykorzystać do instalacji systemu MacOS 10.5
DH4 – ta partycja będzie widoczna pod MacOS jako *Dane* oraz *Użytki* (potem zostanie podzielona pod MacOS na dwie osobne partycje).

Apple Power Mac G5 z monitorem Cinema Display. Kiedyś drogi komputer dla profesjonalistów, dziś tani i wydajny sprzęt dla MorphOS.

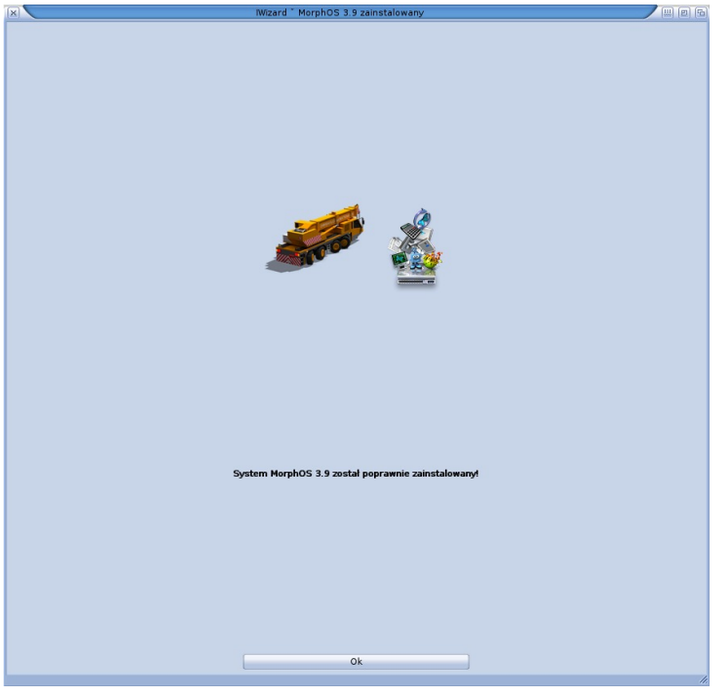


Rysunek 1



Rysunek 2

Rysunek 3



Rysunek 4

Jeżeli ustalimy już sobie rozmiar partycji jaki nas interesuje, to przystępujemy do wyboru miejsca, w którym zainstalujemy MorphOS-a (Rysunek 3).
W naszym przypadku jest to partycja DH1 – nazwana *System* – wielkość 40 GB (39 GB po sformatowaniu).

Po zakończonej instalacji (Rysunek 4) wyjmujemy płytkę z MorphOS-em i wkładamy do napędu płytę z MacOS X (w naszym przypadku MacOS 10.5.1).

Resetujemy komputer, na klawiaturze naciskamy klawisz „C” (wciśnięcie tego klawisza wymusza start z płyty znajdującej się w napędzie) i czekamy aż komputer wystartuje z płyty z MacOS 10.5.1.

Po kilku chwilach powinien pojawić się ekran z wyborem wersji językowej (Rysunek 5).

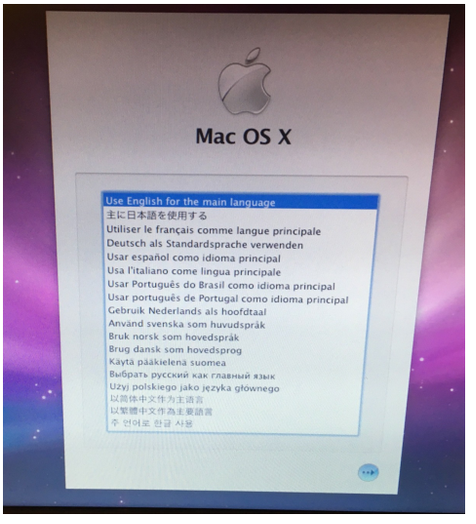
Wybieramy interesującą nas wersję językową (ja wybieram język *Polski*) po czym przechodzimy do następnego ekranu. Na ekranie powitalnym, z górnego menu wybieramy *Narzędzia – Narzędzie dyskowe* (Rysunek 6).

Gdy wybierzemy narzędzia dyskowe, pojawi się okno z wyborem dysków, (*Narzędzie dyskowe* wyświetla wszystkie napędy podłączone do komputera, także DVD, pendrive itd.).

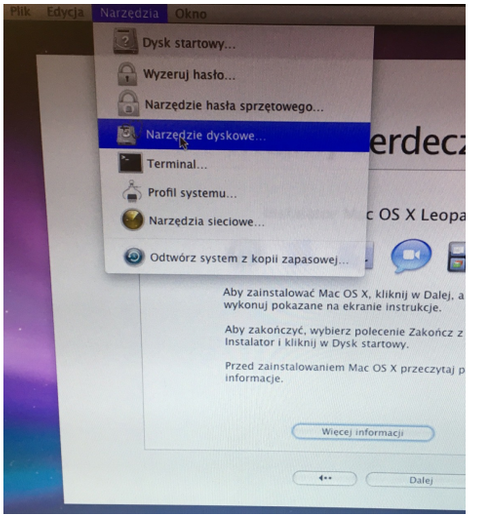
Musimy pamiętać, że ilość partycji o nazwie *disk0s2*, *disk0s3*… itd., może się u każdego z Was różnić. Wszystko jest kwestią podziału dysku, np. na PowerBook G4, podział wygląda jak na obrazku (Rysunek 7).

W naszym przypadku wygląda to tak jak na rysunku (Rysunek 8), gdzie mamy następujące partycje:

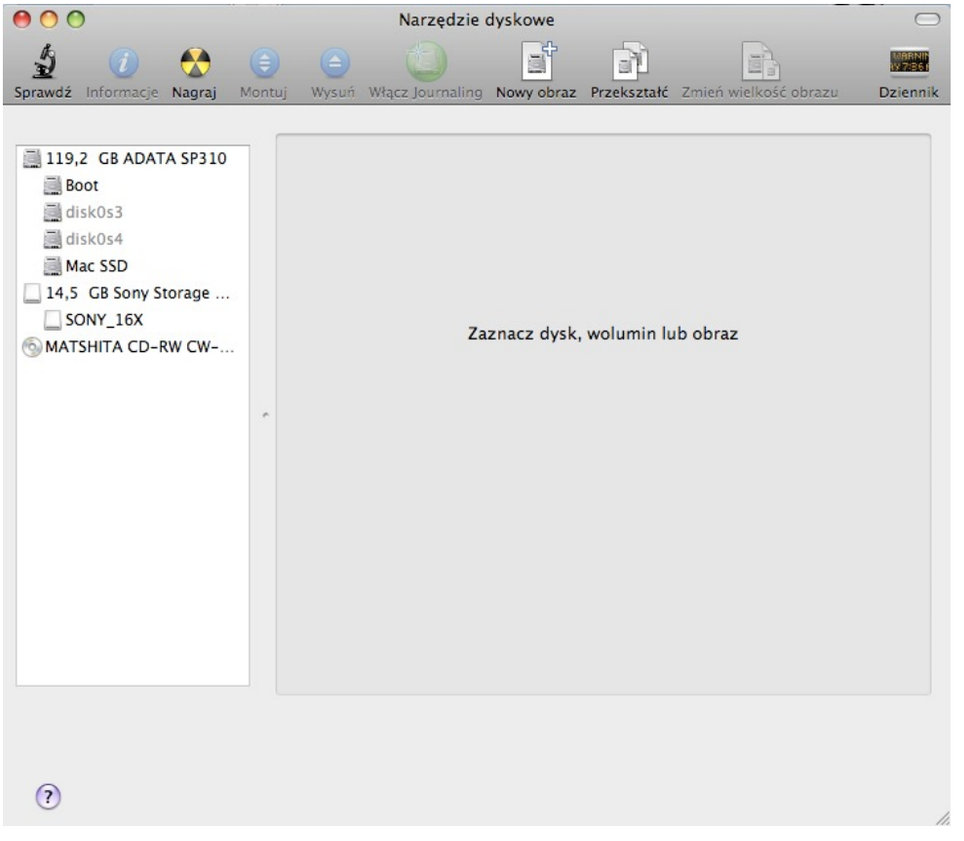
Boot – partycja startowa, którą założył MorphOS – i tego NIE WOLNO nam ruszyć
disk0s3 – to partycja z MorphOS-em – widoczna jako *DH1* (w MorphOS)
disk0s4 – to partycja o nazwie Work – widoczna jako *DH2* (w MorphOS)
Tych partycji nie ruszamy, ponieważ to partycje z MorphOS-em.
disk0s5 – to partycja, na której w naszym przypadku zainstalujemy MacOS 10.5.1
disk0s7 – to partycja na dane, np. filmy, muzykę, gry itd.



Rysunek 5



Rysunek 6



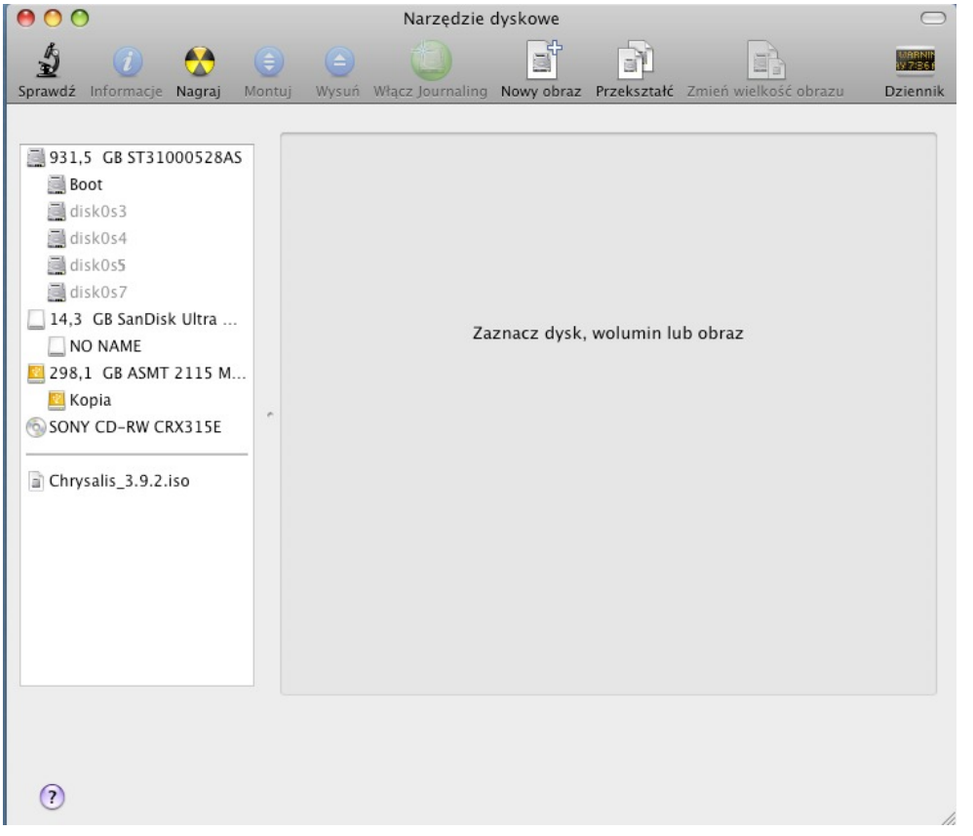
Rysunek 7

Teraz wybieramy partycję *disk0s5* – wybieramy opcję *Wymaż…*. Nadajemy jej nazwę – ja dałem MacOS, następnie formatujemy używając systemu plików Mac OS Extended (Journaled) (Rysunek 9).
Przy okazji możemy podzielić lub sformatować sobie partycję *disk0s7*. Ja podzieliłem ją na dwie kolejne o nazwach *Dane* oraz *Użytki*.
Następnie wychodzimy z Narzędzi dyskowych – klikamy na przycisk *Dalej*.

Pojawi się okno wyboru miejsca, w którym mamy instalować system MacOS, wybieramy nasz dysk, który nazwaliśmy MacOS. Instalacja zabierze kilka/kilkanaście minut w zależności od prędkości dysku, procesora itd. W kolejnym kroku instalacji MacOS, tworzymy konto użytkownika, Apple ID, a potem tylko aktualizacje, konfiguracja… czyli najlepsza zabawa :-).

W przypadku tej metody, system MorphOS będzie tym, który będzie się wczytywał domyślnie – no ale na tym nam chyba zależy :-).
Jeżeli jednak chcemy zalogować się do systemu MacOS, to mamy dwie metody:
– po włączeniu komputera naciskamy klawisz ALT na klawiaturze
– po resecie musimy przytrzymać klawisz ALT

Po chwili pojawi się menu z wyborem ikon systemu (Rysunek 10) – powinny być dwie, jedna z MorphOS, druga z MacOS. Wystarczy wybrać ikonę MacOS i po chwili będziemy cieszyć się systemem Apple.
(Przy wyborze systemu, zazwyczaj trzeba odczekać kilka/kilkanaście sekund, aż będziemy mogli zaznaczyć interesujący nas system, ponieważ OpenFirmware szuka innych napędów/dysków, z których może wystartować).



Rysunek 8

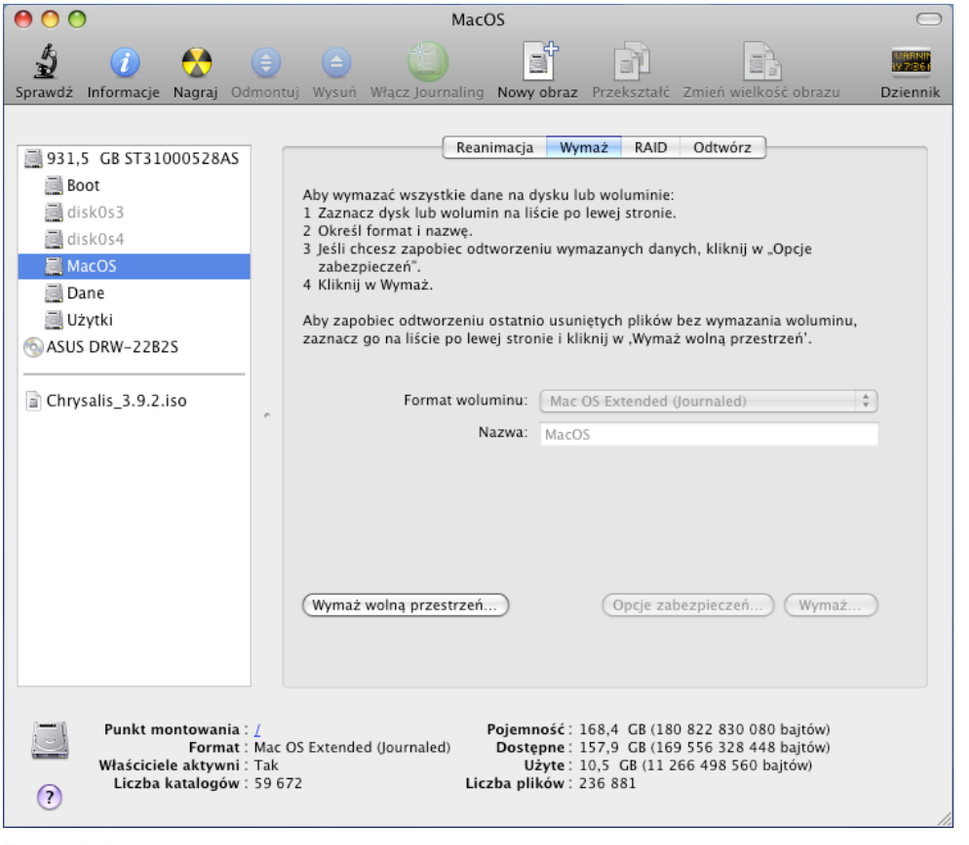
DYSK Z MORPHOS-em

W przypadku gdy mamy już zainstalowanego MorphOS-a, a chcielibyśmy mieć jeszcze MacOS 10.5.8, musimy najpierw przygotować partycję dla MacOS.

Nie jest to takie trudne, do tego zadania wystarczy przecież MorphOS i program HDConfig, który znajduje się na dysku systemowym w katalogu Tools.

Po uruchomieniu HD Tools, pojawi nam się okno (Rysunek 11).

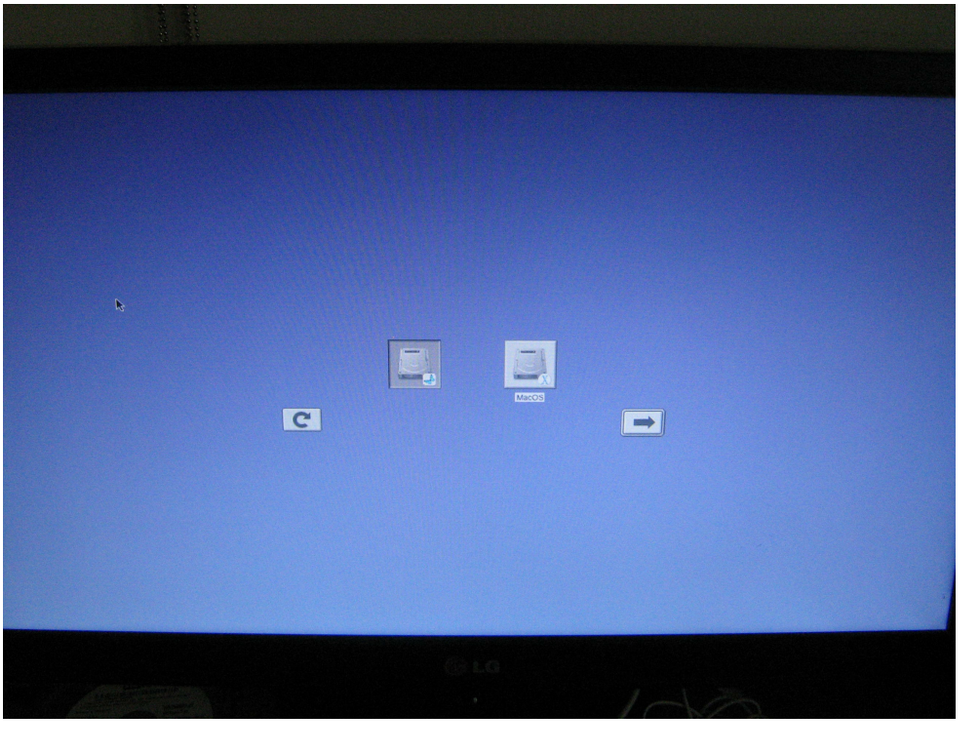
W tym przypadku, największym problemem jest wygospodarowanie na dysku odpowiedniej przestrzeni dla systemu MacOS.



Rysunek 9

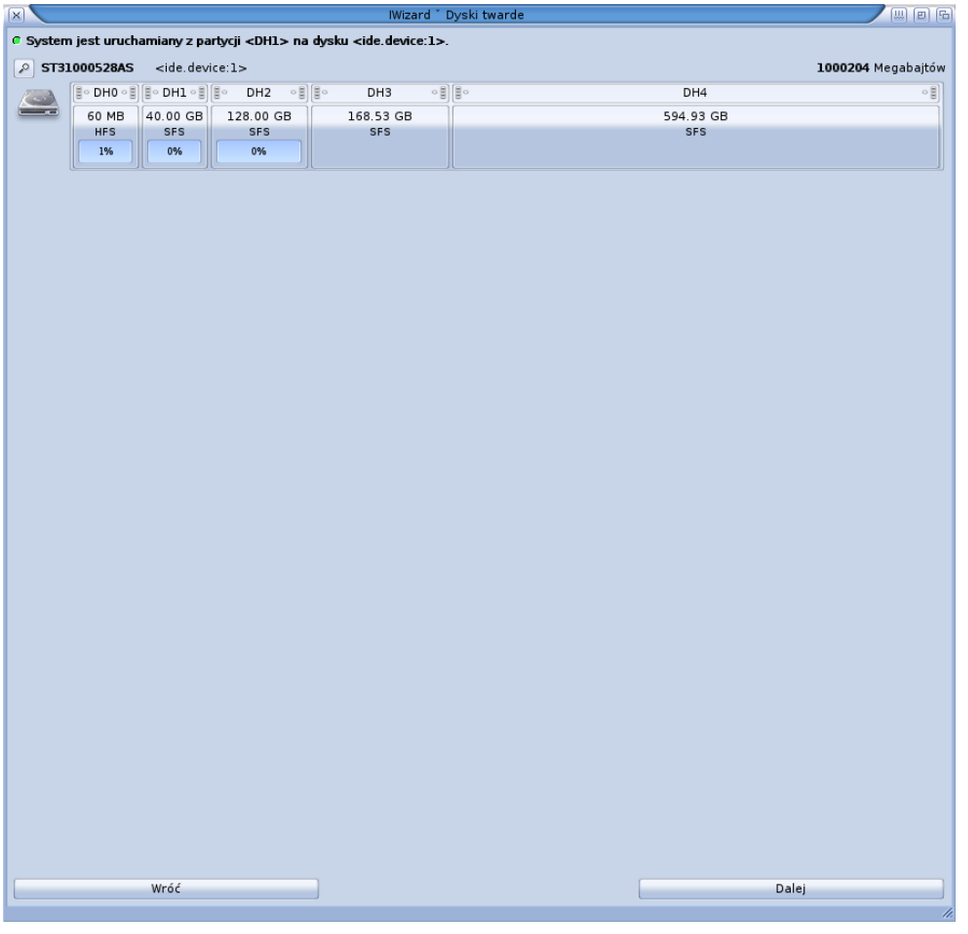
Musimy pamiętać, że jakiekolwiek zmiany na np. partycji *DH4*, doprowadzą do usunięcia z niej wszystkich danych. Warto więc najpierw zrobić sobie kopię bezpieczeństwa danych.

Tak naprawdę, to instalacja MacOS w przypadku gdy mamy już MorphOS-a na dysku, nie różni się niczym od przykładu opisanego powyżej. Jedynym problemem będzie utworzenie partycji. W naszym przypadku przeznaczyłem na MacOS partycje *DH3* i *DH4*. Jeżeli partycja 600 GB jest za duża, można ją podzielić na dwie mniejsze po ok. 300 GB. Można je sformatować, można pozostawić jako ndos, dla instalatora MacOS nie ma to znaczenia, i tak będziemy musieli przeformatować partycje w instalatorze MacOS.



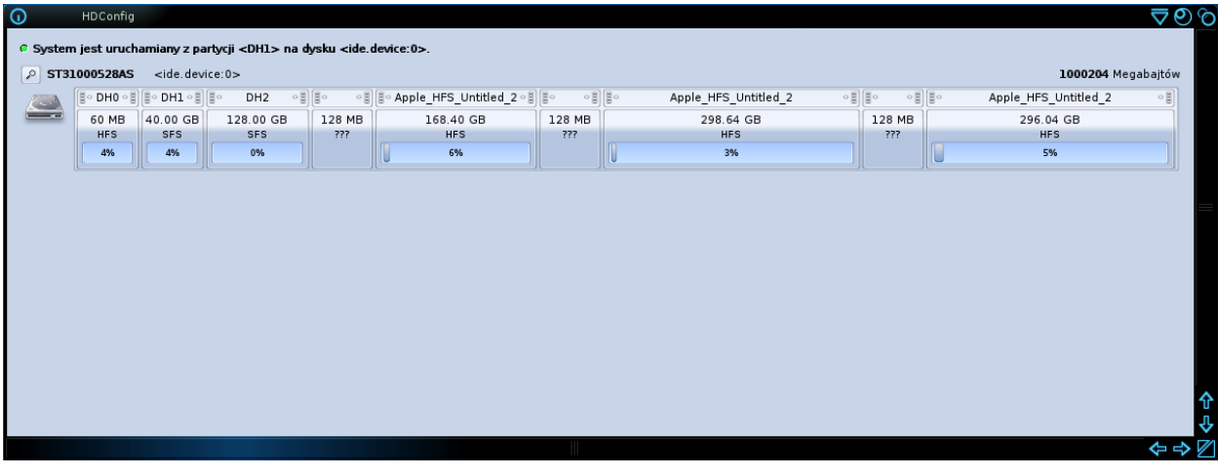
Rysunek 10

Po przygotowaniu naszego dysku, wkładamy do napędu płytę z systemem MacOS i rozpoczynamy instalację systemu wykonując te same kroki opisane powyżej (czyli uruchamiamy narzędzia dyskowe, wybieramy interesującą nas partycję, formatujemy, instalujemy system itd.). Pamiętajmy tylko, że zazwyczaj dwie pierwsze partycje (w naszym przypadku aż trzy, *Boot*, *disk0s3* i *disk0s4*) przeznaczone są dla systemu MorphOS.



Rysunek 11

Po instalacji MacOS, gdy ponownie, z czystej ciekawości, uruchomimy program HDConfig w MorphOS-ie, zobaczymy podobny obrazek (Rysunek 12). Jak widać MacOS 10.5 sam stworzył sobie odpowiednie partycje startowe, buforujące (te o wielkości 128 MB), my tylko musieliśmy podzielić dysk programem HD Config.

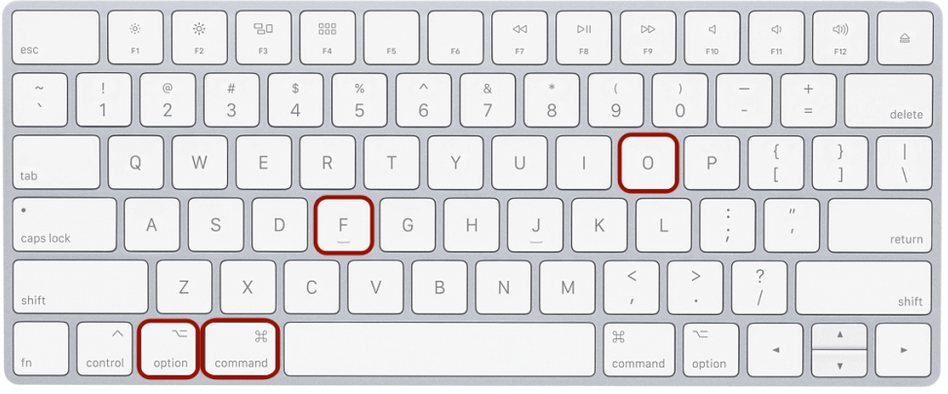


Rysunek 12

Na koniec pozostaje tylko otworzyć Firmware (Rysunek 13) i wpisać komendę:

```
setenv skip-netboot? true
```

Wprowadzenie jej spowoduje, iż podczas startu komputer trochę szybciej zacznie wczytywać system, ponieważ nie będzie szukał systemu w sieci lokalnej.



Rysunek 13

To już wszystko, teraz pozostaje nam tylko zabawa z MorphOS i MacOS.

Tak jak na komputerach AmigaOne można posiłkować się, jako drugim systemem, Linuksem, tak na komputerach Apple naturalną alternatywą dla MorphOS-a jest MacOS X. Więcej o tym – w najbliższym numerze.

Aquaria – inne historie

Phibrizzo

W piątym (2/2011) numerze Polskiego Pisma Amigowego ukazała się recenzja gry przygodowej Aquaria wydanej przez grupę Bit-Blot. Obecnie z tym tekstem można się zapoznać na portalu www.ppa.pl
Gra opowiada historię pewnej morskiej istoty imieniem Naija, w której poczucie samotności wzbudziło chęć odkrycia swojego pochodzenia. Wyrusza w tym celu w przepastne otchłanie oceanu. Napotyka na swej drodze opustoszałe miasta świadczące o istnieniu dawnych cywilizacji oraz dziwne stwory, z którymi będzie musiała się zmierzyć.
I choć gra ma swój klimat to po skończonej grze czuje się lekki niedosyt i każdy chciałby wiedzieć, jak historia potoczyła się dalej.
Autorzy wyszli graczom naprzeciw, pozwalając dołączać do gry swoje dodatki zwane modami.
Takich modów wyszło naprawdę wiele, jednak tylko kilka zasługuje na większą uwagę.
Ostrzegam jednak, że cały poniższy tekst można oznaczyć jako „Uwaga – Spoiler”.

Sequel Mod Chapter One

„– Mój synu, chciałbym ci opowiedzieć historię twojego pochodzenia”. Tak rozpoczyna się pierwszy z opisanych tu dodatków. Jest on opowieścią opowiedzianą przez Li swojemu synowi Lucienowi po szesnastu latach od momentu, gdy Naija zaginęła.
Tu małe wyjaśnienie: główny wątek gry ma dwa zakończenia. Jeśli znaleźliśmy wszystkie tak zwane „Lost Memories” zobaczymy pełniejsze zakończenie, w którym Mia porywa Naiję.

Wracając jednak do historii, Li zaniepokojony tym, że Naija nie wróciła do domu, postanawia ja odszukać. Wybiera się w podróż, gdzie zwiedza nie tylko znane lokacje, ale również całkowicie nowe, w tym bardzo mocno rozbudowaną „Energy Temple”, gdzie dzieje się główna akcja całej historii.
Główny bohater nie potrafi korzystać z mocy Pieśni, a jego jedyną bronią jest strzelająca bransoleta. Jednak korzystanie z niej nie jest takie proste. Strzelając trzeba dobrze wymierzyć gdyż pociski nie są samonaprowadzające tak jak to miało miejsce w głównym wątku.

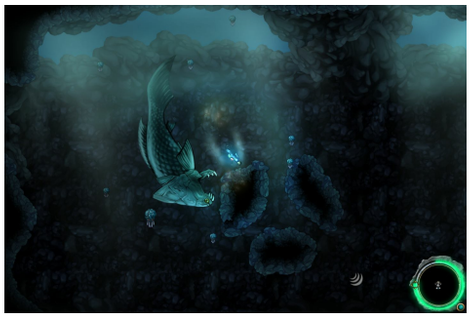
Cala historia nie kończy się niestety happy endem. Autorzy co prawda taki przewidywali jednak postanowili zostawić sobie furtkę by moc rozbić historie na kilka części.
O tym że historia miała być dłuższa można się przekonać przeglądając zawartość plików.
Jak do tej pory dostępna jest tylko pierwsza z nich. Na temat kolejnych jak na razie niewiele wiadomo. Może się kiedyś ukażą.

W samej grze zmieniono fizykę poruszania się postaci na lądzie – Li potrafi wyjść z wody na ląd i poruszać się na dwóch nogach. Na samym początku gry możemy wybrać jeden z czterech stopni trudności.

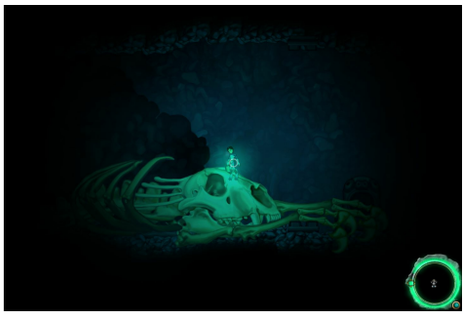
Wysłuchajcie mnie!

Chyba nie mam szans

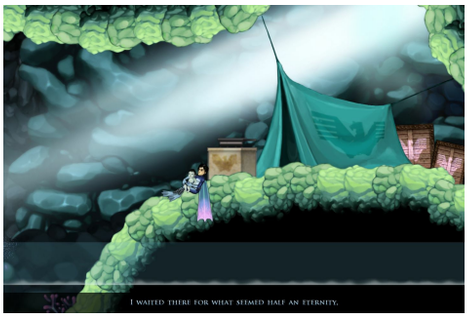
Audiencja u króla



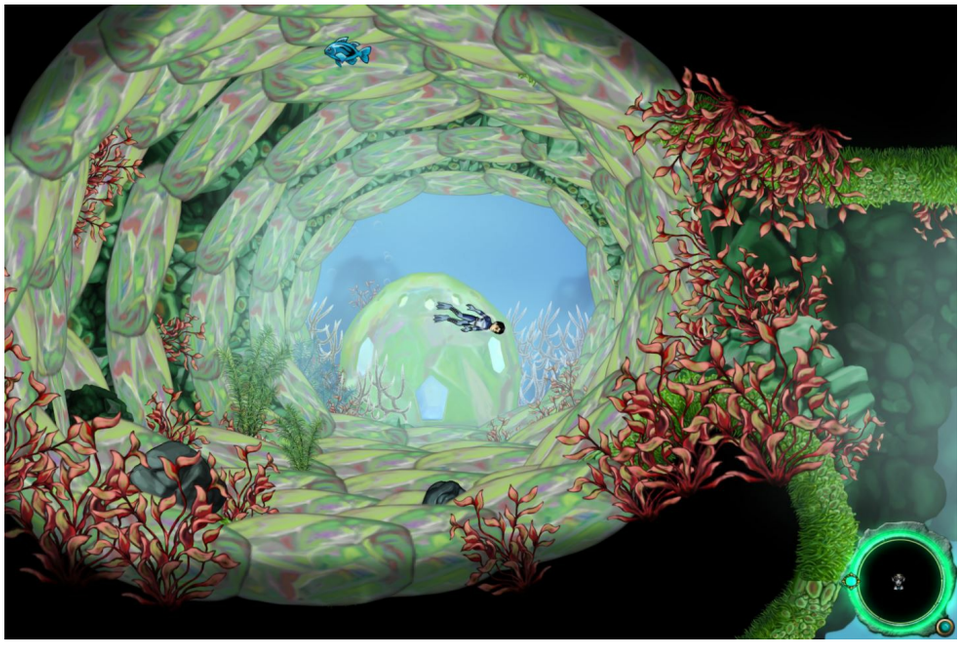
Co to za potwór!



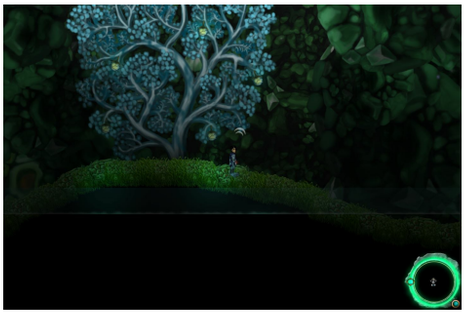
Czyje to szczątki?



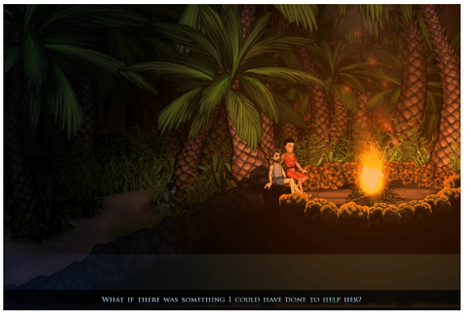
Co ja teraz zrobię..



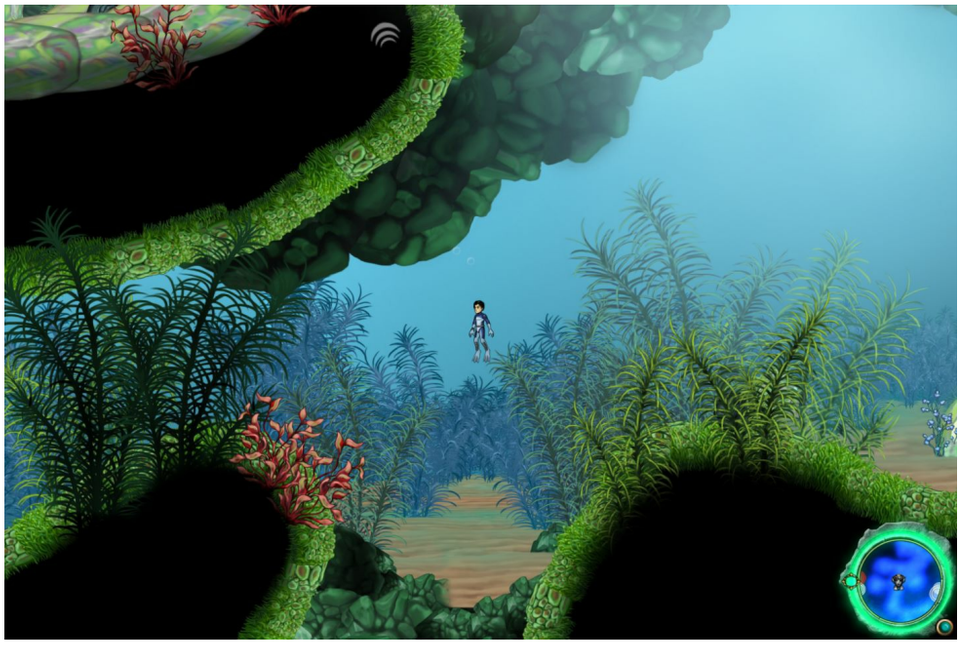
Jak tu pięknie...



Umiem chodzić!



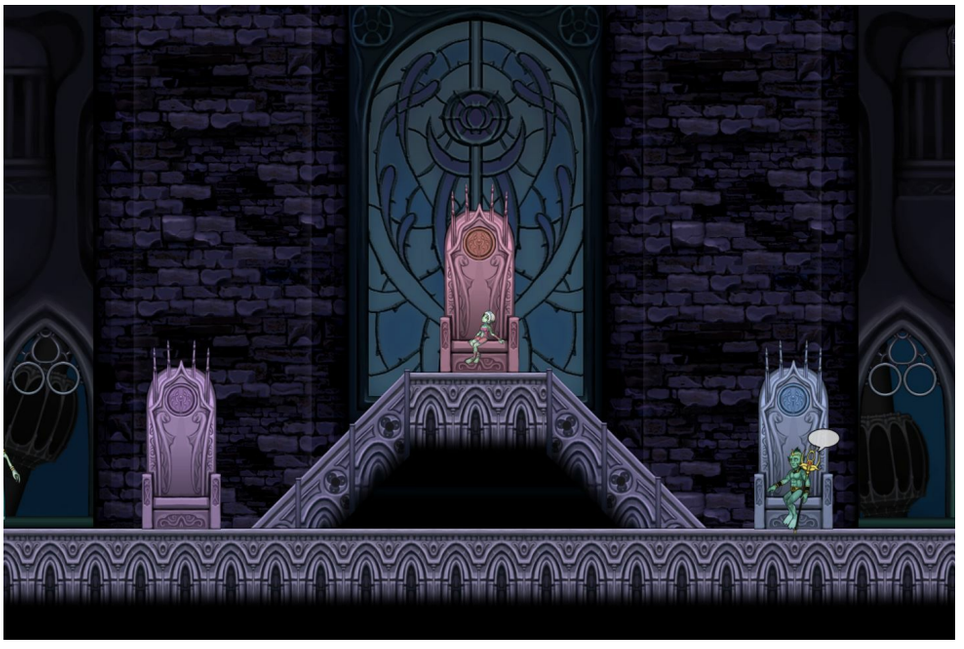
Kto się tam chowa pod palmą?



Którędy teraz?

Kraina Erulian

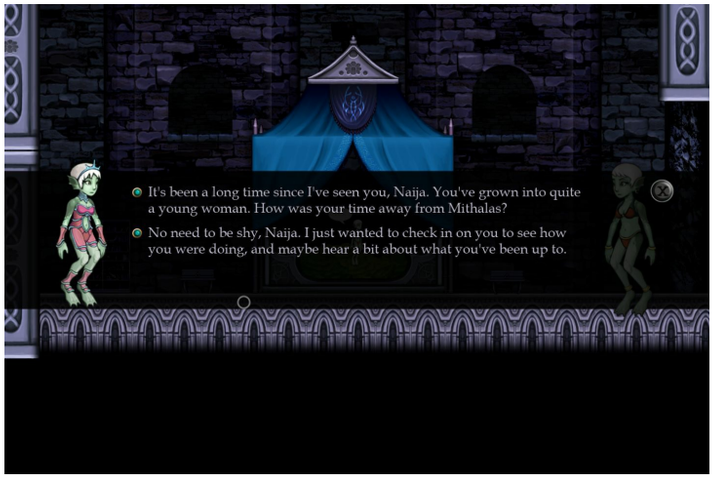
Wojownicy Krotities



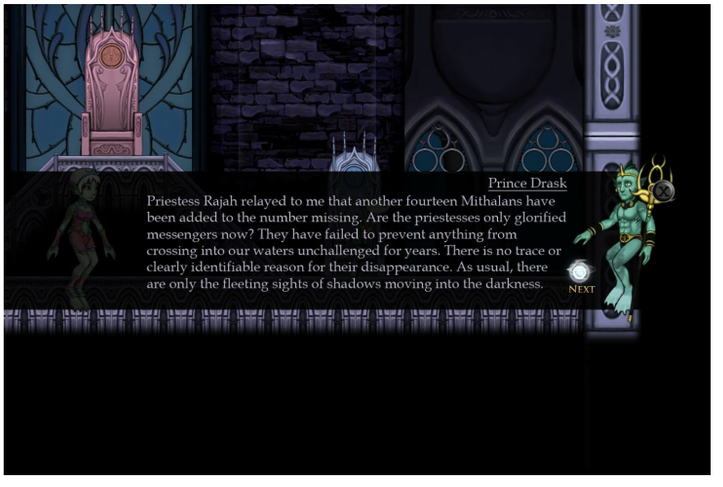
W pałacu na tronie

Sacrifice

Dodatek ten można uważać za prolog do głównego wątku gry. Tym razem przenosimy się w przeszłość. W miecie Mithalas, gdzie mieszkańcy do tej pory pędzili beztroskie życie, zaczęły dziać się niewytłumaczalne rzeczy. Książę Drask zaniepokojony takim stanem rzeczy prosi swoje żonę Elene o kilka przysług, które mogłyby temu zaradzić. Ten dodatek jest prostym RPG-iem w którym musimy wykonać pewne zadania. Nie jest ich dużo a cała gra opiera się na rozmowach z innymi osobami. Spotkamy Naiję będącą jeszcze dzieckiem, porozmawiamy z Mią, dowiemy się kim była „Trzynasta”,



Elena spotyka mala Naije



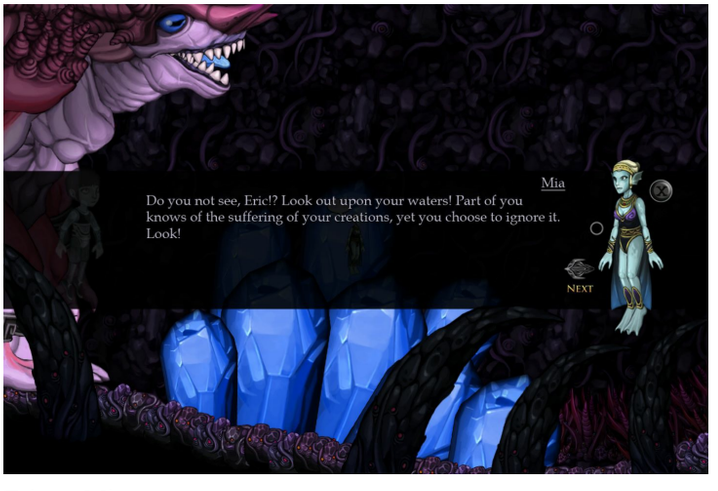
Pogadanka

Stary znajomy

Pogadajmy z Walkerem

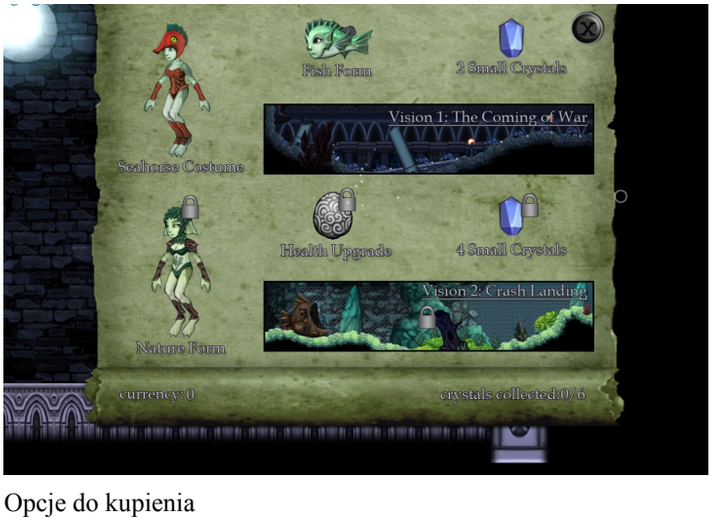
Vedha oraz Sauguin. Dowiemy się również, na czym polegał konflikt Eryka z Mia. W każdym razie nieważne jak taka rozmowa się potoczy, bo i tak nie ma ona wpływu na resztę naszych poczynań. W jednym tylko przypadku odpowiednio poprowadzona rozmowa zapobiegnie walce z jednym z kapłanów.
Oprócz tego w grze znajdują się lokacje, które same w sobie nic nie wnoszą jednak odwiedzenie ich spowoduje, że możemy coś zdobyć lub dowiedzieć się innych informacji. Nowością w tym dodatku jest natomiast to, że podróżując na Żółwiu możemy wybrać konkretne miejsce gdzie chcemy się udać o ile wcześniej to miejsce odwiedziliśmy.

Miasto Mithals

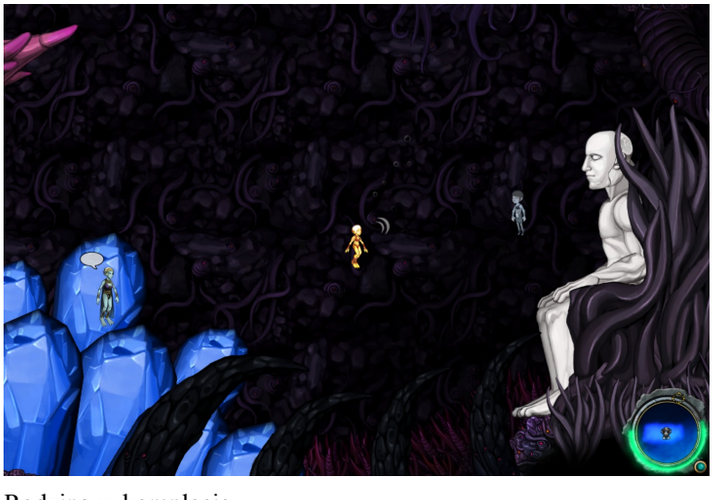


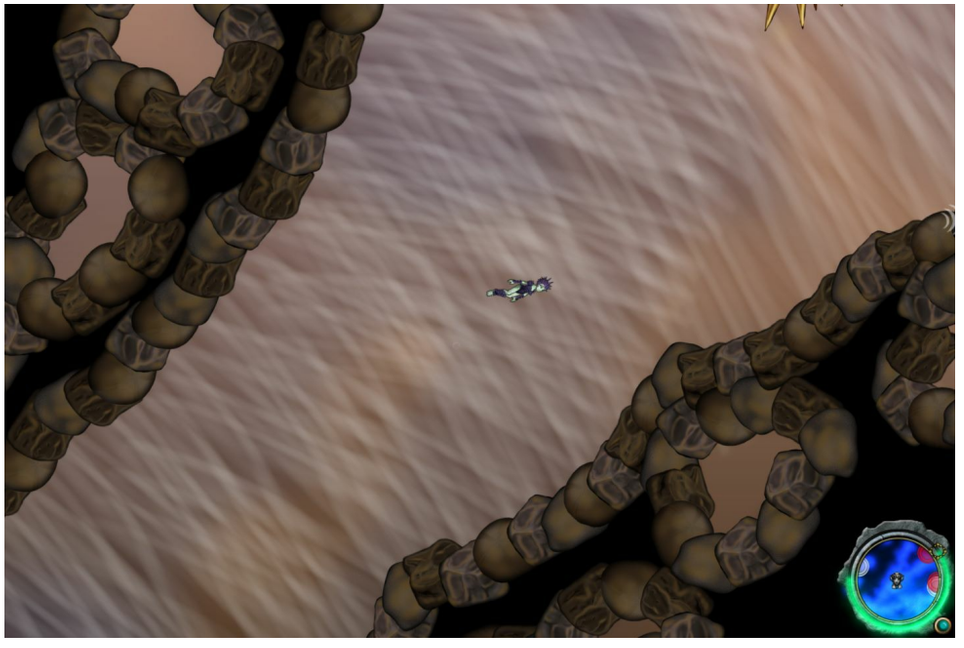
Eric vs Mia

Finałowa walka z Mią



Opcje do kupienia


Rodzina w komplecie



Prądy morskie

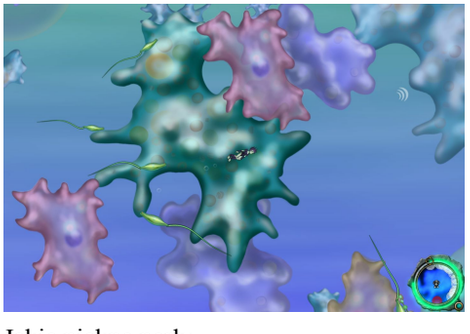
Mamusiu, duchy!

Labyrinth

Ten dodatek nie jest związany bezpośrednio z głównym wątkiem gry. Jest to całkowicie osobna historia dziejąca się po wydarzeniach z głównej części gry.

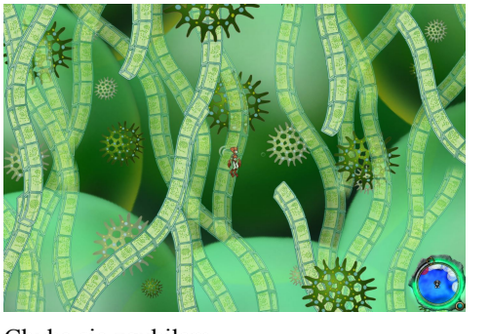
Naija została sama w domu, gdyż Li musiał opuścić wody oceanu, aby rozwiązać kilka spraw.
Nudząc się strasznie postanowiła bliżej przyjrzeć okolicy, w której przyszło jej mieszkać.
Najbardziej zaciekawił ją korytarz zawalony głazem, którego postanowiła usunąć i udać w podróż w nieznane. Na jej nieszczęście wpadła do bardzo rozległego starożytnego labiryntu, z którego nie mogła uciec, gdyż jej droga powrotna została odcięta. Dlatego też musiała poszukać innego wyjścia.

Ten dodatek jest naprawdę rozległy. Autorzy bardzo się napracowali, co widać po lokacjach, które naprawdę zachwycają nową szatą graficzną oraz postaciami, z którymi nigdy nie mieliśmy kontaktu, grając w inne dodatki. Aby utrudnić nieco eksplorację nowych obszarów postanowiono ograniczyć dostęp do map. Jeśli nawet mapa danego terenu jest dostępna to i tak na wiele się nam nie przyda.



Jakie piękne perły

Zielono wokół

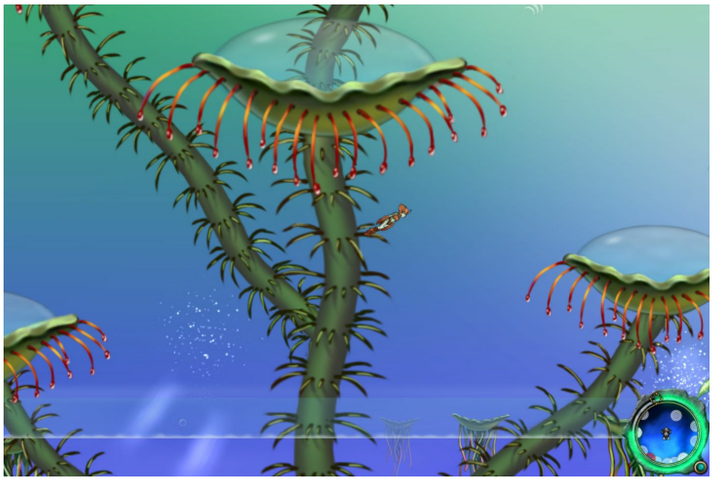


Chyba się zgubiłam

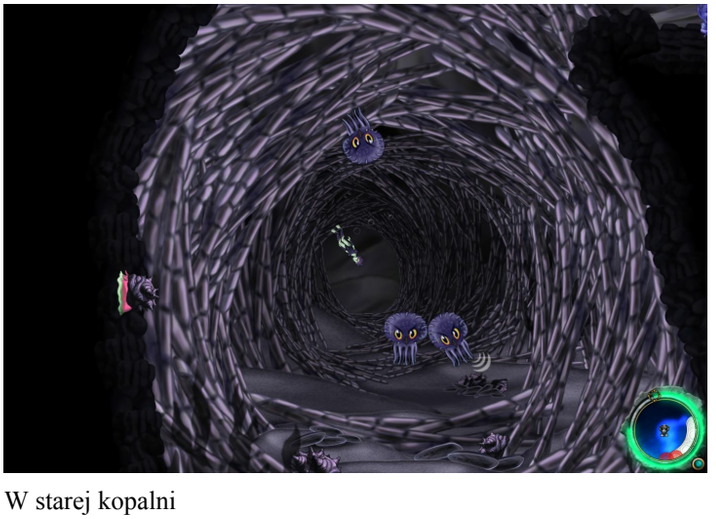
Wystrój krajobrazu, w którym poruszamy się, został tak skonstruowany, że sam w sobie tworzy labirynt. Dodatkowo, oprócz zbierania skarbów znanych głównego wątku, w tym tytule możemy zbierać też tęczowe perły. Na koniec gry zobaczymy statystki związane z naszymi osiągnięciami. Innym ciekawym elementem tego dodatku jest to, że po skończonej eksploracji gra nie kończy się. Gdy Naija wróciła do domu okazało się ze Li tam na nią czekał. Możemy zabrać go ze sobą aby razem wybrać się na wycieczkę po tym tajemniczym świecie.

Z pamiętnika – dzień 5, wyjścia dalej nie widać

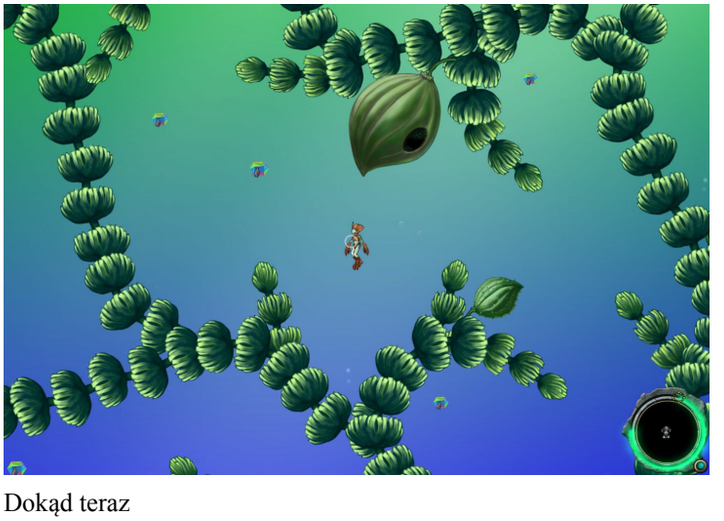
Coś wam zagram



Dziwne rośliny



W starej kopalni



Dokąd teraz

Meatymod

Na sam koniec coś dla ludzi lubiących gry platformowe oraz mających stalowe nerwy.

Pewnego dnia Naija wygrzewała się na powierzchni oceanu zajadając się Rainbow Mushroom. Niestety, ich nadmierna ilość spowodowała że dostała halucynacji. W swym narkotycznym zwidzie zobaczyła jak Li został porwany przez Mię. Jak łatwo się domyślić naszym celem jest uwolnienie go. Jednak nie jest to takie proste.

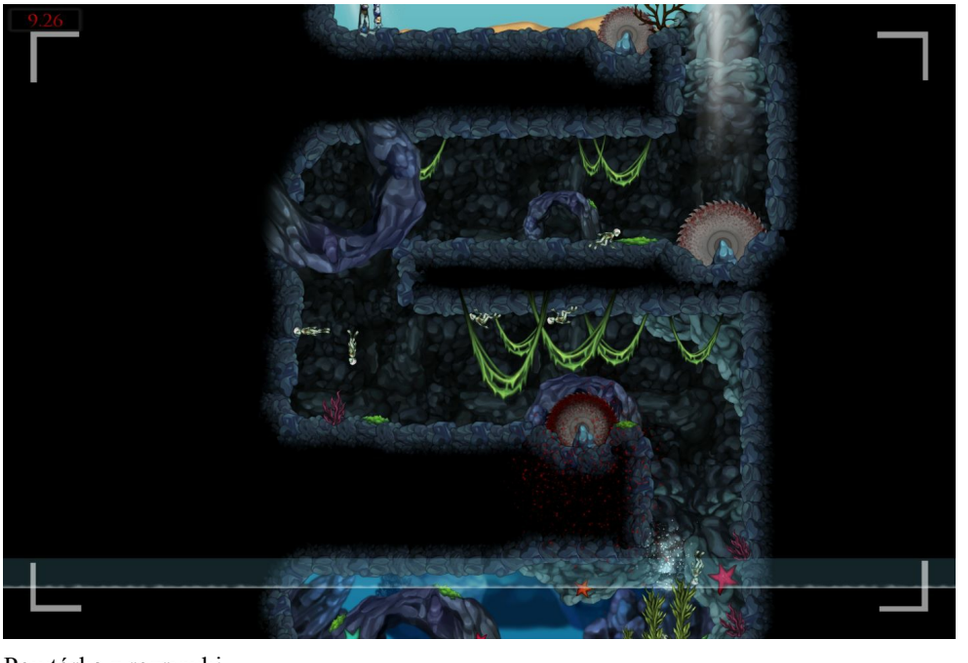
Do przebycia mamy kilkadziesiąt plansz podzielonych na kilka światów. Grając musimy unikać niebezpiecznych elementów, nie tylko opływając je, ale również wspinając się po pionowych ścianach. W całej grze do niczego nie musimy strzelać. Naszymi głównymi przeszkodami są piły tarczowe, które potrafią się poruszać, oraz różnego rodzaju pociski. Rozpoczynając każdy poziom mamy pełen poziom energii, jednak przy spotkaniu z elementem niebezpiecznym natychmiast umieramy w dość spektakularny sposób. Tak, tak, ilość mięsa i krwi przelanej w tej grze przewyższa takie tytuły jak „Doom" czy „Quake". Jednak nie mamy się czym martwić. Posiadamy nieskończoną ilość żyć, więc każdy poziom możemy przechodzić do skutku. Jednak nasze nieudane próby widać na planszy w postaci krwawych śladów, których przy dalszych niepowodzeniach przybywa.

Ciekawym elementem gry jest Replay. Po ukończeniu danego poziomu możemy obejrzeć powtórkę naszych poczynań, tyle że wszystkich naraz. Wygląda to naprawdę ciekawie.

Co za piękny dzień...

Krew się leje

Zostaw go, Wiedźmo!



Powtórka z rozrywki

Instalujemy

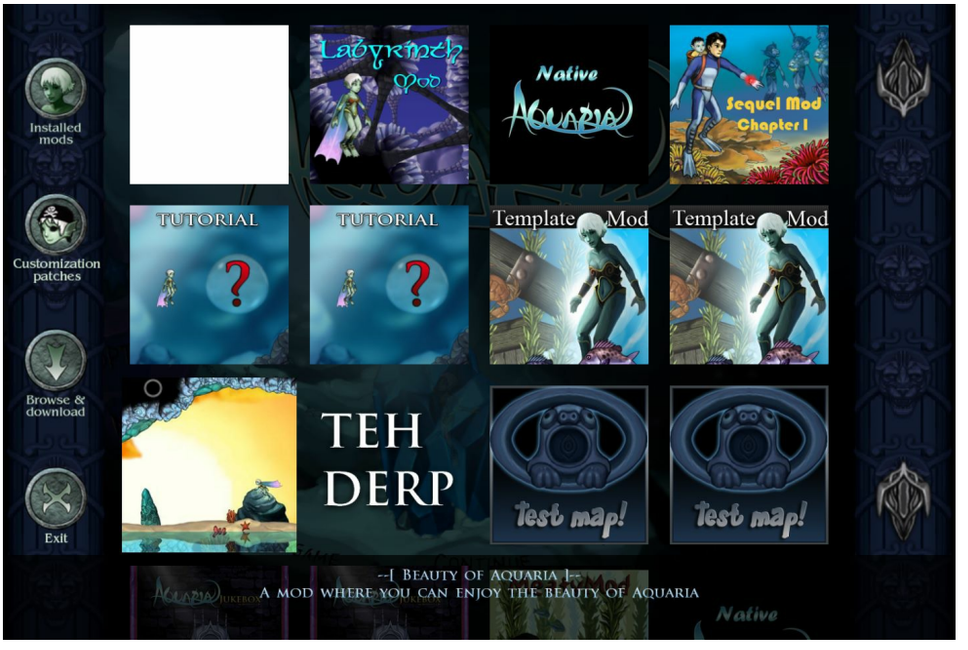
Choć porty gry powstały na systemy takie jak MorphOS czy AmigaOS 4.x, to nie wszystkim będzie dane zagrać w opisane dodatki. Wszystko co tu opisałem testowałem na portach gry powstałych na system MorphOS.

Na starszej wersji portu 1.1.3 „Labyrinth” oraz „Sacrifice” zadziała, ale nie bez problemów. Są one jednak niewielkie.
W „Labyrinth” będzie problem z pokonaniem pierwszego Bossa. Jest to jak najbardziej możliwe, z tym że bardzo dużo czasu będziemy musieli na to poświęcić.
W „Sacrifice” w pewnym miejscu wpadamy w kanał, w którym płynąc towarzyszy nam szum wody. Jednak po wydostaniu się z niego szum w dalszym ciągu jest słyszany, co powoduje dyskomfort. Nie pomaga wyjście do menu głównego. Najlepiej zapisać stan gry, wyjść całkowicie z gry, a następnie kontynuować rozgrywkę od zapisanego miejsca.
Na system AmigaOS 4.x też jest dostępna taka sama wersja portu, dlatego sądzę, że tam też powinny się uruchomić. Możliwe że z podobnymi problemami.

Wszystkie opisane dodatki są darmowe. Można o nich poczytać na oficjalnym forum gry: www.bit-blot.com/forum/index.php w dziale „Modding”.

Jednak ze znalezieniem konkretnych linków do archiwów może być problem. Dlatego aby uprościć poszukiwania podam gdzie je można znaleźć.

Sacrifice
www.moddb.com/mods/sacrifice/downloads/sacrifice



Przeglądarka modów z nowszej wersji gry

Labyrinth
fg.wzff.de/pub/aquaria/mods/Labyrinth_v1.zip

Po ściągnięciu należy je rozpakować i umieścić w katalogu „.Aquaria” znajdującym się w głównym katalogu z grą.
Po uruchomieniu gry wybrać opcję „Mod”, a następnie wskazać to co nas interesuje.

Pozostałe mody, czyli:

Sequel Mod Chapter One
fg.wzff.de/aqmods/aq2_ch1.aqmod

Meatymod
www.moddb.com/mods/meatymod/downloads/meatymod-v102

wymagają nowszej wersji portu opartej na uwolnionych źródłach OSE v1.002, w której rozbudowano silnik gry. Ten port na chwilę obecną istnieje tylko w wersji MorphOS-owej, i można go pobrać stąd:

blabla.ppa.pl/download/Aquaria.lha

Dodatkowo należy doinstalować uaktualnienie plików gry, które również jest darmowe:

fg.wzff.de/pub/aquaria/updatepacks/ose_v1.002/Aquaria_update_win32.zip

W wersji tej rozbudowano manager modów, który praktycznie robi wszystko za nas. Manager pokazuje jakie dodatki są dostępne na sieci, oraz jakie mamy zainstalowane. Jeśli chcemy jakiś dograć wystarczy go wskazać. Gra automatycznie połączy się z serwerem, pobierając go i instalując w odpowiednim miejscu, co powoduje że możemy go natychmiast uruchomić.

Na koniec

Jeśli komuś spodobały się pierwsze przygody Naiji, to powinien sięgnąć po dalsze tytuły, właśnie w takiej kolejności, w jakiej tu przedstawiłem.

Wspomnień czar starego dziadka – od Amigi do MorphOS-a

trOLLO

Amiga 500 (1989-1994)

Amigę mam od 1989/90. Najpierw była to Amiga 500, którą dostałem od rodziców w nagrodę za dobre oceny w szkole.
Dlaczego Amiga? Bo wtedy nikt innego komputera nie kupował, był to jedyny rozsądny wybór jeżeli chciało się grać. Wszyscy w okolicy mieli Amigi 500, wszyscy na nich grali, od każdego można było pożyczyć gry.

Jednak to co MNIE urzekło w tym komputerze to system oraz demoscena – chociaż wtedy nie wiedziałem, że to tak się nazywa. Demoscenę poznałam dzięki koledze z klasy w szkole podstawowej o imieniu ADAM – jeżeli to czyta – to pozdrawiam!
System AmigaOS 1.3 był dla mnie czymś niesamowitym w porównaniu do MS-DOS jakiego używało się wtedy na większości PC które widziałem. Jako, że komputer był często używany do oglądania dem, zastanawiałem się jak można by coś w nim zmienić, tak aby wszystko chodziło lepiej. W 1992/93 roku, Amiga 500 została uzbrojona w 20 MB dysk twardy i 2 MB RAM – wtedy też pierwszy raz obejrzałem demo – Odyssey grupy Alcatraz. Dostałem to demo od kolegi Adama na kilku dyskietkach.

Amiga 1200 (1994-2004)

Amigę 1200 kupiłem w gliwickim Domu Handlowym „IKAR”.

Komputer posiadał wbudowany dysk 40 MB, jak dla mnie był to gigant i za bardzo nie wiedziałem jak ten dysk zapełnić. Pamiętam, że po zakupie Amigi 1200, przez długi czas siedziałem i zastanawiałem się jak podzielić dysk na partycje, 40 MB dysku to przecież bardzo dużo. Oczywiście pierwszą rzeczą jaką zrobiłem, po zakupie Amigi 1200 i podzieleniu dysku na partycje, było uruchomienie wspomnianego już dema Odyssey, musiałem zobaczyć jak będą działały animacje, które na Amidze 500 nie były zbyt płynne.

Amiga 1200 bardzo szybko dostała dodatkowe 4 MB Fast RAM i zaraz po dołożeniu RAM-u uruchomiłem demo Nexus – 7, które wcześniej nie chciało mi działać. Sam dysk twardy został wymieniony na 100 MB, ponieważ oryginalny dysk zamontowany w Amidze po dłuższym czasie okazał się jednak bardzo mały, te wszystkie dema scenowe zajmowały coraz więcej miejsca.
Wymagania rosły (głównie przez dema scenowe) i Amiga szybko dorobiła się karty z MC68030 – był to Blizzard IV, który został wymieniony na Elboxa 1230 (w sumie nie pamiętam dlaczego Blizzarda wymieniłem na Elboxa). Jako, że uwielbiałem oglądać dema scenowe, to po zakupie karty z MC68030 pierwszym demem jakie uruchomiłem było Voyage in Storm.

Przyczyną mojego bardzo szybkiego upgrade'u Amigi były:
– Demoscena
– Shapeshifter

Dema scenowe wymagały szybkiej wymiany procesora, po prostu na MC68030–40 MHz + FPU większość dem działała źle, szczególnie dema grupy TBL (The Black Lotus). Dlatego też bardzo szybko kupiłem MC68040–33 (kręcone na 40 MHz).

Ze Shapeshifterem było podobnie. Na MC68030 niektóre gry działały fatalnie, wymiana MC68030 na MC68040 spowodowała, że takie gry jak Settlers II, Doom, Sim Tower mogły działać w miarę płynnie na Amidze.

Podczas modernizacji Amigi ok. 1998 r. wystąpiła bardzo śmieszna sytuacja. Otóż mój kuzyn był zapalonym użytkownikiem PC i fanem Windowsa (dzisiaj nazwalibyśmy go fanboyem). Natomiast ja, jako posiadacz Amigi 1200, często nabijałem się z jego PC i problemów jakie miał. Pewnego pięknego wakacyjnego dnia, razem wybraliśmy się po zakup dysku twardego – Seagate 3.2 GB ATA 33, ja do swojej Amigi 1200, on do swojego PC – wtedy chyba jakiegoś Celerona/Pentiuma 266 MHz. Śmieszna sytuacja polegała na tym, że na mojej zabawkowej Amidze (jak to kuzyn nazywał), podział dysku na partycje, instalacja systemu i przegranie danych ze starego dysku (wtedy ok 200 MB HD), zabrały ok. 1 godziny. Kuzyn po dwóch dniach bojów z dyskiem zrezygnował z instalacji dysku – komputer wprawdzie widział go, ale tylko 2 GB. Do dzisiaj mu to wypominam w różnych naszych komputerowych „sporach".

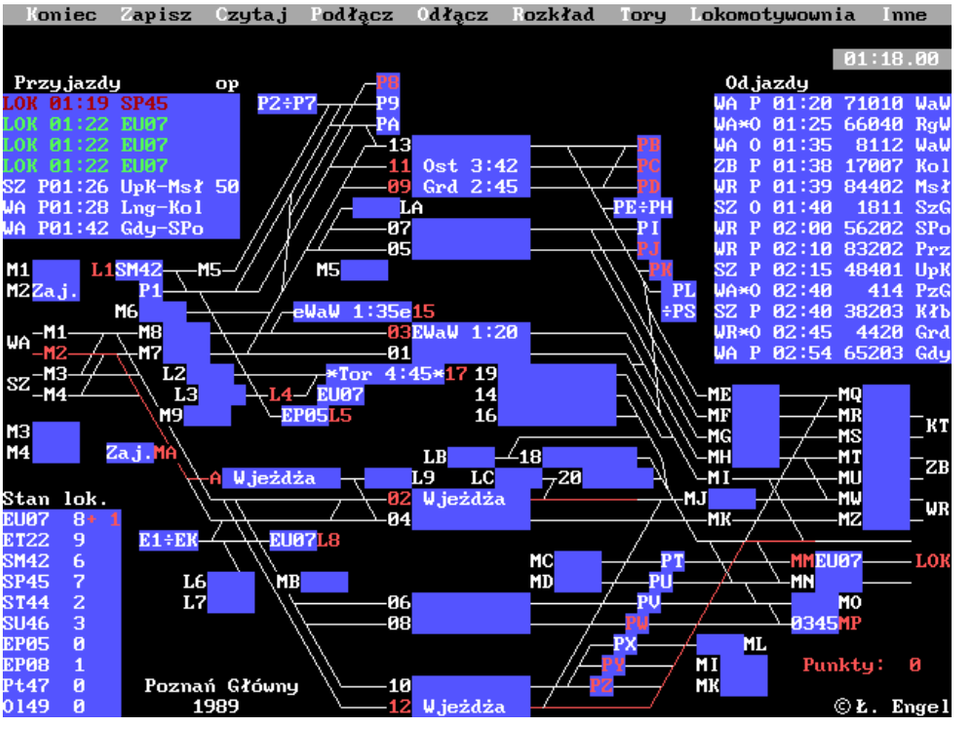
Końcówka lat 90. to już Amiga 1200 w E/Box, z Fast/ATA i procesorem MC68040–40 MHz, która była używana do gier i szkoły. To na niej pisałem pracę dyplomową w, nieistniejącym już, Technikum Kolejowym w Gliwicach no i oczywiście używałem do oglądania dem scenowych.

Takie dwie ciekawostki:

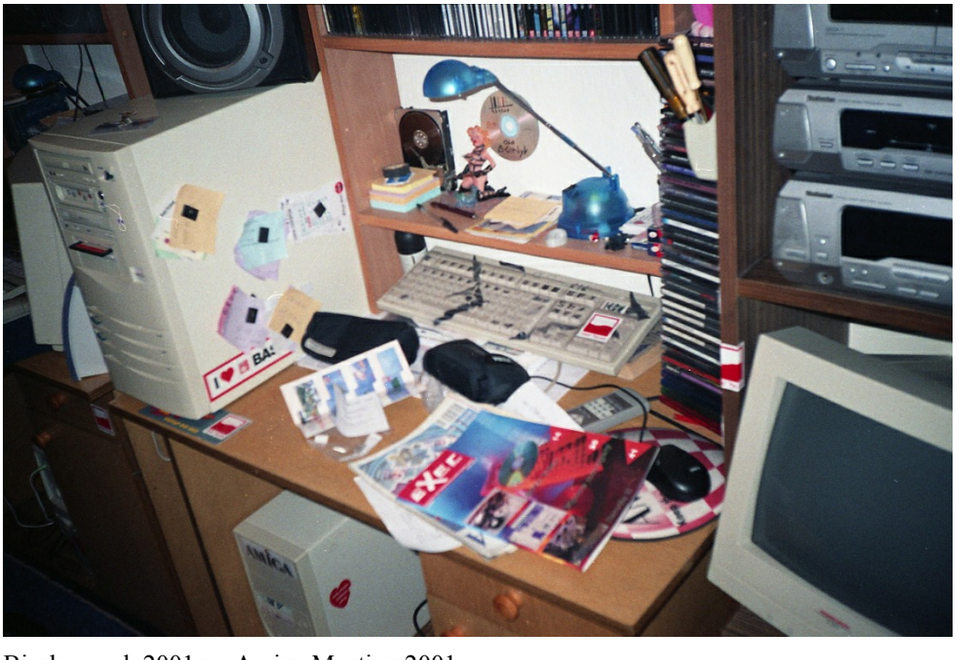
Koledzy, często mnie odwiedzali, aby na A1200 w programie WordWorth składać i drukować prace dyplomowe. Kilku z nich miało nawet PC, ale twierdzili, że na moim komputerze działa to wszystko zdecydowanie lepiej i płynniej.

W piątej klasie Technikum, na dniach otwartych naszej szkoły, pokazywałem na Amidze 1200 pod PC-Taskiem symulator prowadzenia ruchu kolejowego.
Dlaczego tak? Ano dlatego, że w tamtych czasach archaiczne 386, w które wyposażona była nasza szkoła, nie za bardzo nadawały się do pokazów. Poza tym, symulator ten był częścią mojej pracy dyplomowej.

Największym moim problemem były awarie sprzętu wywołane przeze mnie – przez moją głupotę. Pamiętam jak w okolicach 2001/02 roku, jadąc ze swoją Amigą 1200 na jedno ze spotkań GGUA, zamówiłem kartę graficzną z Efunzine (chyba z tej firmy). Karta przyszła do mnie na



Symulator ISDR

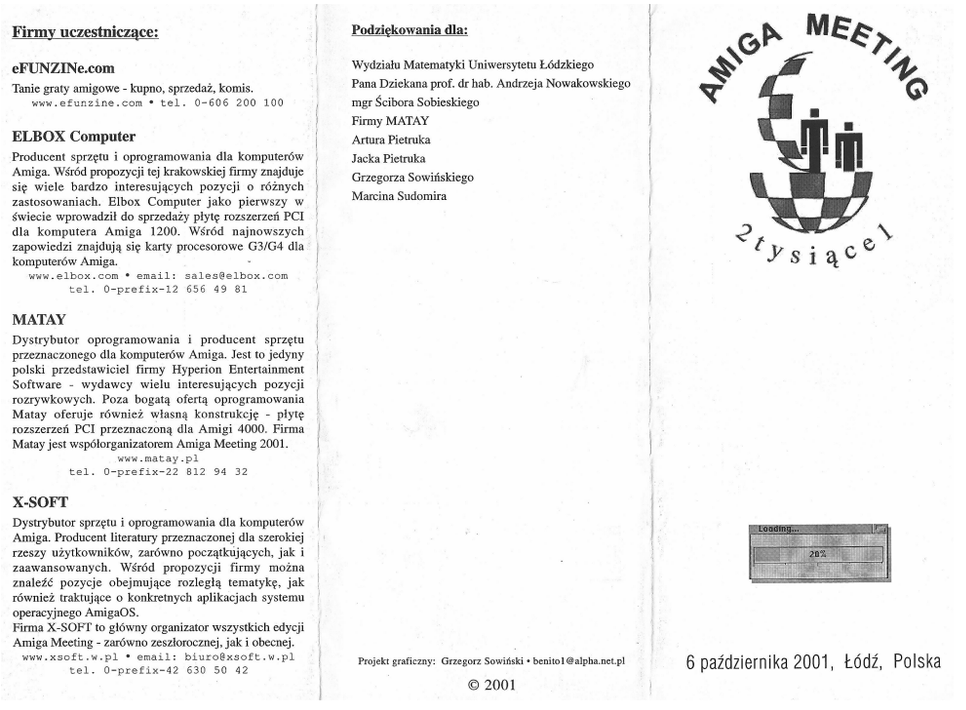

Biurko – rok 2001 po Amiga Meeting 2001

kilka dni przed spotkaniem. Zadowolony, że pochwalę się chłopakom nowym nabytkiem – spaliłem kartę poprzez złe włożenie kości pamięci do niej (była to już chyba trzecia spalona karta graficzna). Na spotkaniu GGUA nawet nie wspomniałem jaki ze mnie był fajtłapa.
Na szczęście w okolicy 2002/03 moja Amiga 1200 miała sprawną kartę graficzną Picasso II, kupioną w nieistniejącym już komisie w Gliwicach na ulicy Szarej – w tamtych czasach niemal kultowe miejsce. Zawsze spotkało się tam Amigowca, zawsze można było porozmawiać, i zawsze były fajne „graty" do Amigi za dobre pieniądze.

Ostateczna konfiguracja mojej Amigi 1200 (w 2002/03 r.), na której pisałem pracę inżynierską na Politechnice Śląskiej, to:

– MC68040–40 MHz (przez krótki okres miałem 68060–50 MHz, ale karta spaliła się)
– Zorro II
– Picasso II +
– Fast/ATA
– 20 GB HDD
– CD-R/RW
– drukarka HP 920 c
– monitor SVGA LiteON 17"

Tę konfigurację mam do dzisiaj, z tą drobną różnicą, że jest tam dysk 32 GB SSD i napęd DVD, a stary monitor SVGA został wymieniony na nowoczesny LCD 21".



Firmy uczestniczące:

eFUNZINe.com
Tanie graty amigowe - kupno, sprzedaż, komis.
www.efunzine.com • tel. 0-606 200 100

ELBOX Computer
Producent sprzętu i oprogramowania dla komputerów Amiga. Wśród propozycji tej krakowskiej firmy znajduje się wiele bardzo interesujących pozycji o różnych zastosowaniach. Elbox Computer jako pierwszy w świecie wprowadził do sprzedaży płytę rozszerzeń PCI dla komputera Amiga 1200. Wśród najnowszych zapowiedzi znajdują się karty procesorowe G3/G4 dla komputerów Amiga.
www.elbox.com • email: sales@elbox.com
tel. 0-prefix-12 656 49 81

MATAY
Dystrybutor oprogramowania i producent sprzętu przeznaczonego dla komputerów Amiga. Jest to jedyny polski przedstawiciel firmy Hyperion Entertainment Software - wydawcy wielu interesujących pozycji rozrywkowych. Poza bogatą ofertą oprogramowania Matay oferuje również własną konstrukcję - płytę rozszerzeń PCI przeznaczoną dla Amigi 4000. Firma Matay jest współorganizatorem Amiga Meeting 2001.
www.matay.pl
tel. 0-prefix-22 812 94 32

X-SOFT
Dystrybutor sprzętu i oprogramowania dla komputerów Amiga. Producent literatury przeznaczonej dla szerokiej rzeszy użytkowników, zarówno początkujących, jak i zaawansowanych. Wśród propozycji firmy można znaleźć pozycje obejmujące rozległą tematykę, jak również traktujące o konkretnych aplikacjach systemu operacyjnego AmigaOS.
Firma X-SOFT to główny organizator wszystkich edycji Amiga Meeting - zarówno zeszłorocznej, jak i obecnej.
www.xsoft.w.pl • email: biuro@xsoft.w.pl
tel. 0-prefix-42 630 50 42

Podziękowania dla:

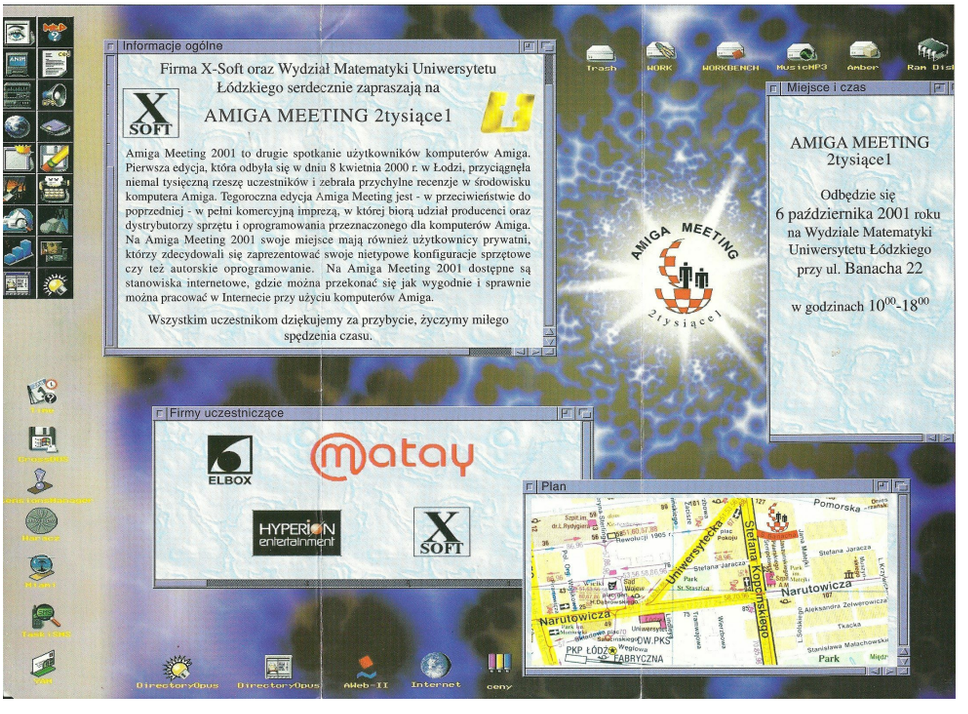
Wydziału Matematyki Uniwersytetu Łódzkiego
Pana Dziekana prof. dr hab. Andrzeja Nowakowskiego
mgr Ścibora Sobieskiego
Firmy MATAY
Artura Pietruka
Jacka Pietruka
Grzegorza Sowińskiego
Marcina Sudomira

AMIGA MEETING 2tysiące1

Loading... 28%

Projekt graficzny: Grzegorz Sowiński • benito1@alpha.net.pl
© 2001

6 października 2001, Łódź, Polska

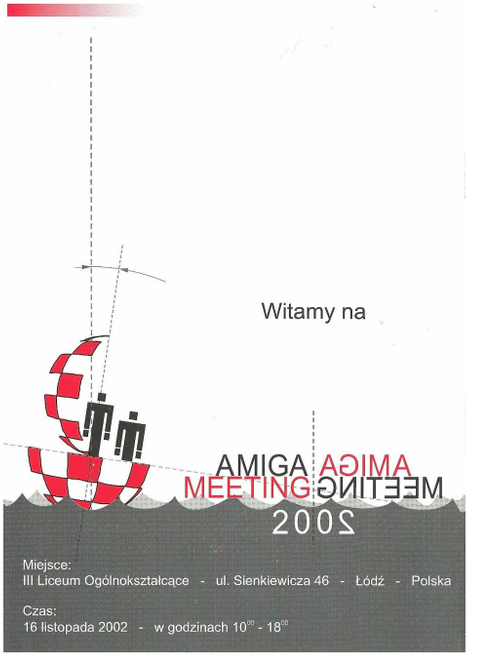


Studiując na Politechnice Śląskiej w Gliwicach, na pierwszym i drugim roku mieliśmy przedmiot – informatyka – który tak naprawdę ograniczał się do nauki obsługi Windowsa, Worda i innych głupot. Jako dumny Amigowiec, posiadacz Amigi 1200 z 040–40 i kartą graficzną, wspomniałem o pewnych zaletach AmigaOS w porównaniu do Windowsa 98. Wykładowca powiedział, że jak przyniosę ten komputer i pokażę mu to o czym mówię, to do od razu zaliczy mi całą informatykę (wszystkie trzy semestry), przyniosłem, pokazałem i zaliczenie dostałem. Trzy semestry informatyki miałem zaliczone zaraz na początku mojej nauki na „Polibudzie".

Pegasos II / I (2005-2007)

Niestety czas mijał a nowej Amigi nie było, a wysłużona MC68040–40MHz sprawiała wrażenie coraz wolniejszej.

Cały czas szukałem alternatywy dla mojej Amigi 1200 i niestety coraz częściej spoglądałem na PC. W 2001 roku na Amiga Meeting po raz pierwszy usłyszałem, plotki, o nowym komputerze, który ma działać na nowej wersji systemu „amigowego", natomiast sam komputer ma być wyposażony w procesor PowerPC, szybkie AGP itd. Niestety, poza niesprawdzonymi informacjami, nie wiedziałem nic więcej, ale była to doskonała informacja dla osoby takiej jak ja, która szukała powoli zamiennika wysłużonej Amigi 1200.



Rok później, w 2002 r., podczas Amiga Meeting w Łodzi, zaprezentowano Pegasosa I z MorphOS-em. Wspaniały, nowoczesny komputer, wyposażony w piękny system, który od razu mi się spodobał. Jednak system MorphOS miał wtedy poważne problemy – wieszał się, był bardzo niestabilny, no ale jednak był to nowy system „amigowy", na nowym, szybkim, i w miarę nowoczesnym sprzęcie. Niestabilny system trochę mnie przerażał (pamiętam, że kilka razy zawiesiłem MorphOS-a poprzez np. otwarcie napędu CD) ale stwierdziłem, że przecież to tylko system, a wszystko można naprawić za pomocą aktualizacji.

Na Amiga Meeting 2002 jechałem z zamiarem kupna Pegasosa I (nikomu nigdy o tym nie mówiłem), niestety nie udało mi się go dostać, czego, paradoksalnie, bardzo mocno żałowałem i jednocześnie się cieszyłem, wszak niestabilny system nie sprawiał za dobrego wrażenia. Z racji braku sensownej alternatywy, na początku 2004 r. kupiłem PC, jednak zakup PC nie spowodował braku zainteresowania Amigą, wręcz przeciwnie, stwierdziłem, że MUSZĘ mieć nową Amigę lub Pegasosa.

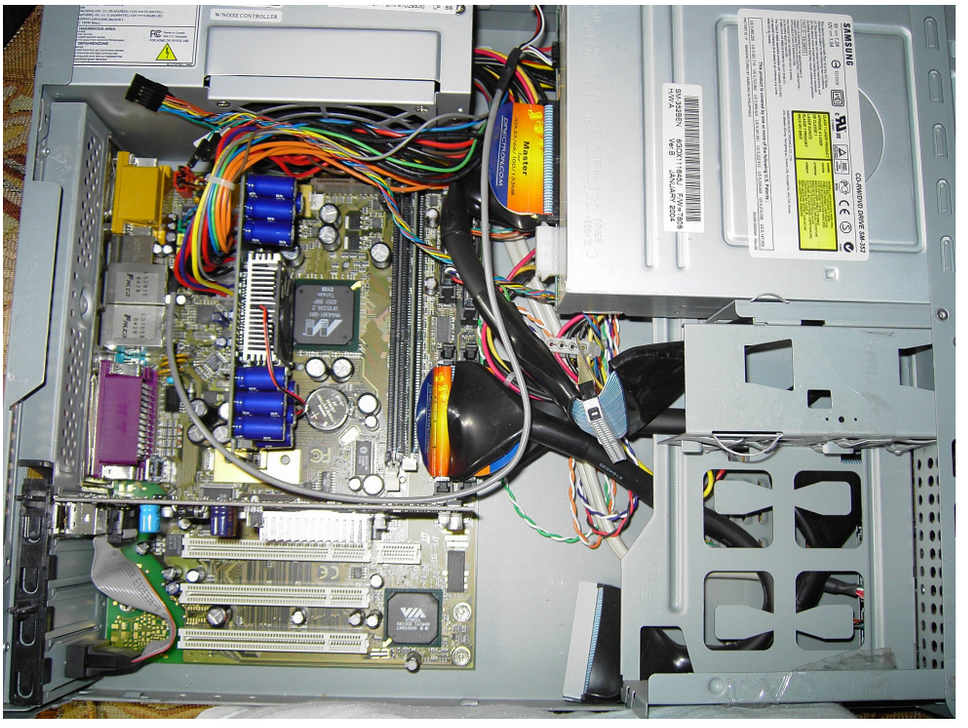
Zainteresowałem się też bardzo mocno AmigąONE z AmigaOS4, jednak tutaj problemy sprzętowe (pierwsze modele miały problemy z IDE, USB, muzyką itd.), brak skończonego OS4 (pamiętam jak zobaczyłem po raz pierwszy AmigaOS 4, który działał wolniej od systemu na mojej Amidze 1200), i ciągłe przenoszenie premiery spowodowały, że MorphOS, w moim odczuciu, był lepszym rozwiązaniem.

Pegasos II – MorphOS 1.45

Pod koniec 2005 r. stałem się posiadaczem Pegasosa II – wprawdzie pożyczonego, ale jednak byłem posiadaczem. Pegasos II miał następującą konfigurację:

– Procesor PPC G4 – 1 GHz
– RAM 512 MB
– Radeon 9250 64 MB VRAM
– HDD 4 GB

Po wielu bojach z konfiguracją systemu, wiedziałem, że MorphOS, będzie moim docelowym systemem „amigowym", ale brak dobrej przeglądarki www spowodował, że nie rozważałem w tamtym okresie zakupu tego komputera.
Pegasosa II niestety musiałem oddać, więc szukałem jakiegoś w miarę taniego rozwiązania, aby znowu używać MorphOS-a. Na przełomie 2007/08 roku stałem się, na bardzo krótko, (nie) szczęśliwym posiadaczem Pegasosa I (w wersji bez April, dostałem tylko płytę z procesorem, obudowę wykorzystałem z PC, którego sprzedałem), więc dla mnie praca z MorphOS-em przypominała raczej drogę przez piekło.

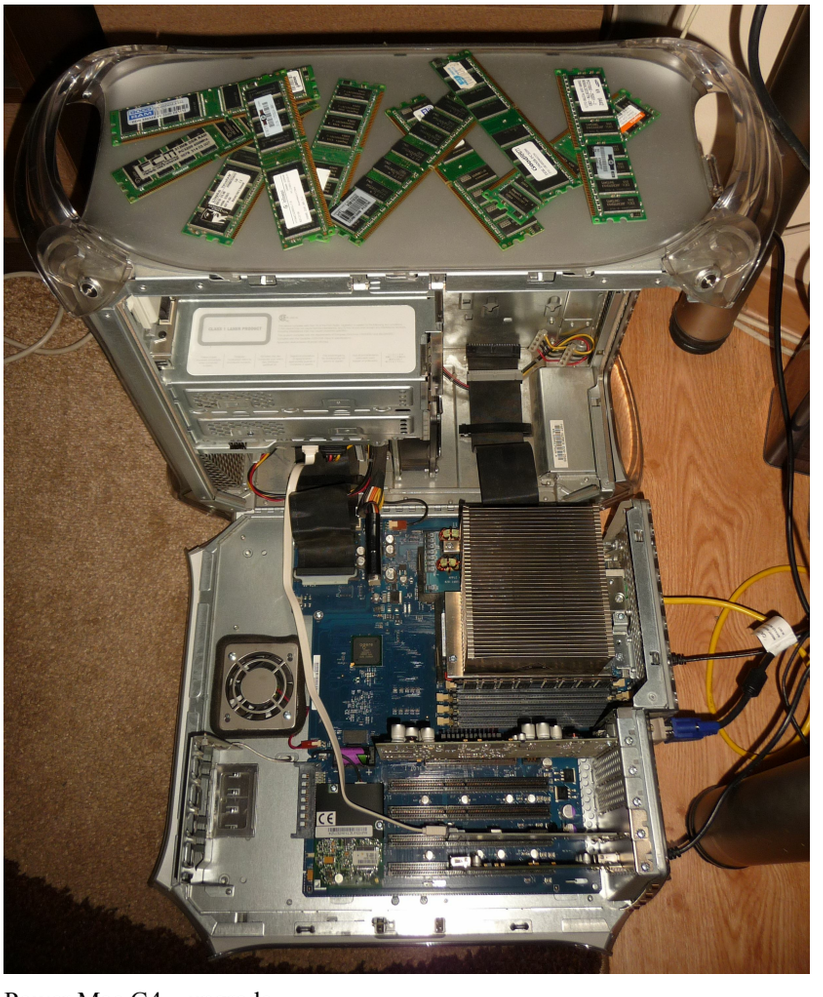


Pegasos II

Power Mac G4/G5 (2010–….)

W okolicach 2010/11 kupiłem sobie Power Maca G4 – 867 MHz (jeżeli mnie pamięć nie myli to była taka konfiguracja), na którym zainstalowałem MorphOS-a 2.7 – wtedy jeszcze wersje DEMO, dalej brak sensownej przeglądarki powodował, że nie widziałem sensu zakupu systemu za dosyć duże pieniądze jak na system hobbystyczny (wtedy koszt MorphOS-a to 111 Euro).

Tak naprawdę to „zabawa na całego" zaczęła się w ok. 2013 roku wraz z zainstalowaniem MorphOS-a 3.1, wtedy na Power Mac G4 – 900 MHz, 512 MB RAM, ATI Rage



Power Mac G4 – upgrade

128 PRO – 16 MB VRAM (karta szybko została wymieniona na Radeona 7500 32 MB VRAM, a ta bardzo szybko na Radeona 9000 PRO 64 MB VRAM). Od tego czasu komputer z MorphOS-em jest w zasadzie moim drugim głównym komputerem, na którym często robię wszystko.
Po złożeniu Power Maca G4, zainstalowaniu MorphOS-a, na dysku wylądowało całe oprogramowanie z Amigi 1200. Pierwsze parę tygodni spędziłem na graniu w Exodusa, Quake I, Payback, Return to Castle Wolfenstein i inne. Oczywiście nie mogło zabraknąć demosceny, więc na dysku znalazły się wszystkie możliwe dema jakie tylko znalazłem na PowerPC i MorphOS-a.

Obecnie posiadam następujące komputery z MorphOS:

Power Mac G4 MDD – 1.25 GHz
– RAM 2 GB DDR
– Radeon 9000 PRO 128 MB VRAM
– HD 1 TB SATA II

Power Mac G5 2 x 2.0 GHz
– RAM 4 GB DDR
– Radeon 9600 XT 128 MB VRAM
– HD 120 GB SSD, 1 TB HDD

PowerBook G4 – 1.5 GHz
– RAM 1 GB DDR
– Radeon 9700 M – 64 MB VRAM
– HD 120 GB SSD mSATA

Do niedawna Power Mac MDD G4 (w różnych konfiguracjach), był moim głównym MorphOS-owym komputerem. Na nim przeglądałem www, odbierałem pocztę e-mail, oglądałem filmy, słuchałem muzyki- itd. Czasami uruchamiałem także MacOS 10.5.8 gdy zachodziła potrzeba np. skanowania dokumentów lub napisania jakiegoś tekstu w Wordzie – no ale o tym wkrótce.

Od niedawna przenoszę wszystkie dane na Power Mac G5, który powoli staje się głównym komputerem do wszystkiego (mimo iż obok stoi „wypasiony" iMac z Core i5 i 16 GB RAM).
Ktoś może zapytać – dlaczego używasz takiego złomu, skoro obok masz nowego iMaca. Odpowiedź jest zawsze prosta – BO LUBIĘ.
Systemu Amigi używam od dawna i jest on dla mnie czymś zupełnie naturalnym, zresztą, przecież do słuchania ulubionej muzyki wcale nie potrzeba Core i5 i 16 GB RAM.

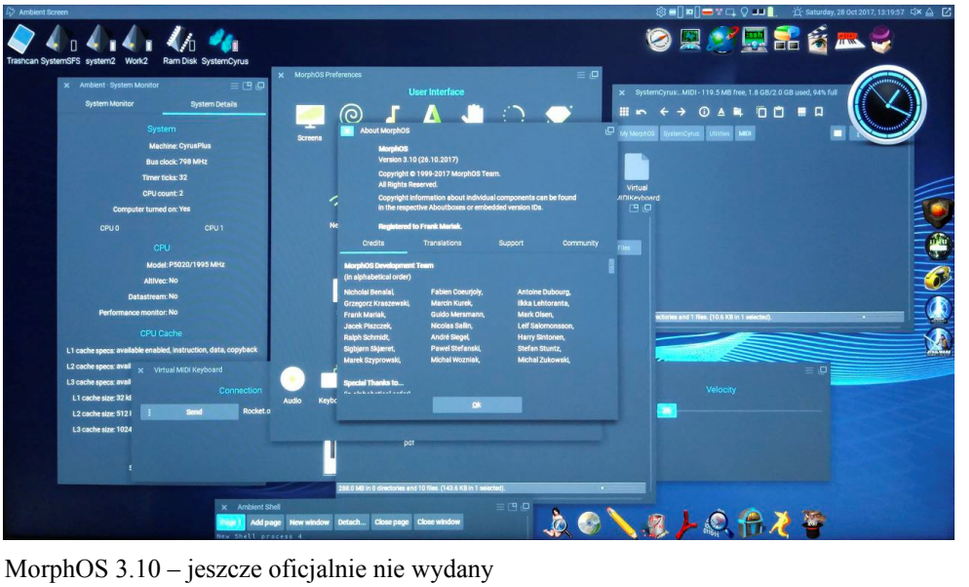
MorphOS jest w zasadzie bezproblemowy w użyciu i bardzo szybki w porównaniu do MacOS 10.5.8, owszem ma swoje humory, niedoskonałości i braki, ale w ogólnym rozrachunku jest to całkiem dobry system. A jego niedociągnięcia, cóż miejmy nadzieję, że wszystko w przyszłości zostanie rozwiązane poprzez stosowne aktualizacje.

Oczywiście to nie wszystkie komputery jakie posiadałem, miałem też Amigę CD32 z ProModule, Amigę 2000 z 030–33 MHz i dyskiem 500 MB, kilka Power Maców G4, jednego Power Maca G3 itd.

Wiem też, że pewnie pomieszałem pewne daty i zdarzenia, ale tyle tego było na przestrzeni ostatnich prawie 30 lat, że można się troszkę pogubić.

Ogólnie, Amigi i podobnych do niej komputerów używam od 1989 r. Uważam, że komputer ten miał w swoim czasie najlepszy system operacyjny jaki mógł powstać. W dzisiejszych czasach Amiga to doskonałe hobby pozwalające odpocząć od wszechobecnych Windowsów, Androidów, macOS i innych cudactw.

Mam nadzieję, że moja zabawa z MorphOS, Amigą i może kiedyś z AmigaOS4 będzie trwała jeszcze wiele, wiele lat. Czego sobie i Wam życzę.



MorphOS 3.10 – jeszcze oficjalnie nie wydany

Prezenty dla PowerMac G4/G5

trOLLO

Święta, święta, i po świętach.
Z okazji świąt postanowiłem powymieniać w swoich komputerach kilka części.

PowerMac G4 MDD

W tym komputerze postanowiłem wymienić dysk twardy, ponieważ obecny dysk 1 TB powoli sugerował, że wkrótce uda się na emeryturę (znikające dane, błędy odczytu itd.).

Największym problemem były złącza IDE stosowane w Power Macu G4 MDD. Dyski IDE obecnie są trudno dostępne i drogie, dlatego w tym modelu mam kartę SATA PCI opartą na chipie SIL3512 (po przeflashowaniu karta idealnie działa pod MorphOS i MacOS).

Do Power Maca G4 wybrałem dysk Seagate Barracuda ST1000LM048 (dysk 2.5” 1 TB 128 MB Cache 5400 obr.).
Ktoś może się dziwić dlaczego dysk 2.5”? Odpowiedź jest prosta – był bardzo tani (139 zł na dzień 28.12.2017).

Prosty benchmark pod MacOS (Xbench) twierdzi, że dysk ma prędkość odczytu ok. 90-100 MB/s, co dla MorphOS-a jest chyba wartością jak najbardziej wystarczającą. Dodatkową zaletą jest jego bezgłośna praca (mój G4 jest wyciszony, nie mam wyjących wentylatorów w zasilaczu).

Drugim zakupem były dwa nowe napędy DVD – ASUS DRW-22B2S

jeden dla Power Mac G4 drugi dla G5. Model ten jest w wersji IDE, i bez problemu działa z jednym i drugim komputerem.
Napęd zapisuje idealnie płyty DWD-R/RW/DL, odczytuje bezproblemowo nawet stare płyty CD wypalane jeszcze na A1200 w okolicach 2000 r.

Power Mac G5 Dual

W przypadku tego komputera największym wyzwaniem jest bardzo wybredny kontroler SATA I (SATA 150) . Kontroler ten w zasadzie nie działa z dyskami SSD SATA III, (SSD SATA II działają bezproblemowo). Mój obecnie używany dysk SSD (GoodRam 120 GB CX100) przestał poprawnie działać, tzn. mogłem odczytać dane z dysku, ale próba nagrania czegokolwiek powodowała zawieszenie systemu, zarówno MorphOS jak i MacOS, dlatego postanowiłem dołożyć drugi dysk 1 TB do tego komputera (obecnie zamiast tandemu SSD+HDD są dwa HDD 1 TB).

Wybór padł na dysk WD10EZEX-75WN4A0 (dysk 3.5” SATA III 1 TB 64 MB Cache 7200 obr.). XBench pod MacOS pokazuje odczyt na poziomie ok. 120-130 MB/s. Dysk jest cichy, działa bezproblemowo z moim PowerMac G5.
W przypadku MorphOS nie widać różnicy w prędkości wczytywania

systemu lub OWB w stosunku do SSD, natomiast MacOS 10.5 Leopard, uruchamia się trochę dłużej w porównaniu do SSD .

Jako, że miałem możliwość przetestowania kilku dysków w G5 przed zakupem, poniżej jest krótka lista dysków, które przetestowałem.

Następujące dyski SATA III działają pod PowerMac G5:

- Seagate Barracuda ST1000LM048 – 1 TB SATA III – 128 MB Cache
- WD10EZEX-75WN4A0 – 1 TB SATA III – 64 MB Cache
- TOSHIBA DT01ACA2 – 2 TB SATA III – 64 MB Cache
- Seagate ST2000DM006 – 2TB SATA III 64 MB Cache

Dyski, które nie działają z Power Mac G5 :
- WD ST3000DM008 – 3TB SATA III – 64 MB Cache
- WD RED WD30EFRX – 3TB SATA III – 64MB Cache
- SSD GoodRam SSDPR-CX300-120 – 120 GB SATA III
- SSD SanDisk SDSSDA-120G-G25 – 120 GB SATA III

W przypadku SSD SATA III nie znalazłem jeszcze dysku, który poprawnie działałby z Power Mac G5; natomiast dyski „talerzowe" nie sprawiają problemów, pod warunkiem, że nie wybierzemy dysku większego niż 2 TB. Przypuszczalnie jest to spowodowane tym iż wg. Apple, kontroler SATA w G5 widzi max. 2 TB.

Power Mac G5, oprócz nowego dysku na święta dostał jeszcze nową mysz. Jest to model Logitech MX Anywhere 2 z nadajnikiem micro-USB.

Oczywiście działa bezproblemowo zarówno z MorphOS jak i MacOS.

Z drobnostek jakie jeszcze kupiłem z myślą o wykorzystaniu z komputerami to:

- dysk USB 3.0 STDR1000200 – 1 TB dysk zewnętrzny 2.5" – służy mi jako kopia bezpieczeństwa ważnych danych (zdjęcia , mp3, pliki Word itd.); bezproblemowo działa pod MorphOS

- pendrive USB 3.0 32 GB D33021 – model z firmy Sony działa z MorphOS

- czytnik kart SD/MicroSD Unitek Y-9321 – działa bezproblemowo z MorphOS

Oczywiście PowerMac G5 nie posiada USB 3.0, więc wszystkie urządzenia działają z prędkością USB 2.0 – średni odczyt z dysku USB i Pendrive Sony to ok. 22 MB/s .

Wszystkie opisane powyżej dyski i pendrive podpinałem do swojego Power Maca G5 (identyfikator Apple to Power Mac 7,3 wersja bootrom 5.2.4f1), dlatego może się zdarzyć, że to co u mnie działało na innym modelu nie zadziała i odwrotnie, to co mi nie zadziałało na innym modelu zadziała.

Podsumowując, to były udane prezenty świąteczno-noworoczne dla Power Maca i MorphOS-a.

Hollywood APK Compiler czyli zróbmy sobie APKę

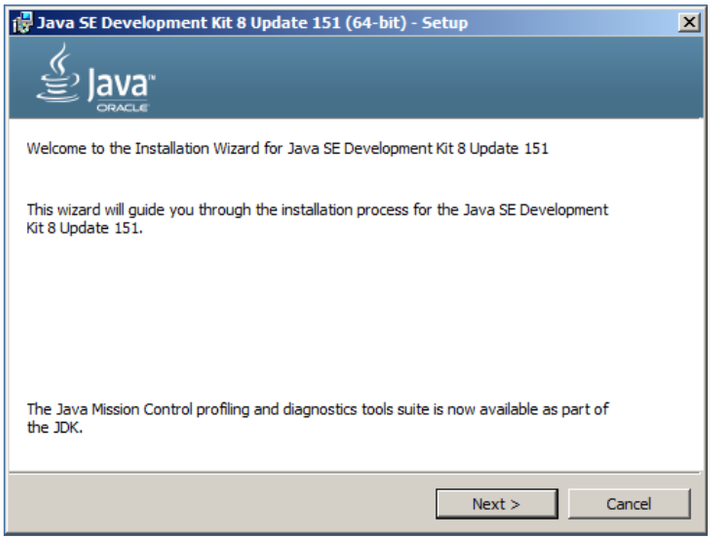
recedent

Jedną z wielu interesujących funkcji pakietu Hollywood jest możliwość kompilowania swoich skryptów na popularną platformę Android (poczynając od archaicznej wersji 2.3.3 Gingerbread wzwyż), co daje możliwość uruchamiania ich na najpopularniejszych smartfonach, czy tabletach. Sam proces kompilowania pliku .apk (czyli gotowego do instalacji na urządzeniu pakietu) nie jest skomplikowany, ale przygotowanie sobie środowiska do kompilacji to nie jest — delikatnie mówiąc — droga usłana różami. W tym artykule postaram się przybliżyć temat z perspektywy laika, którym sam byłem próbując zrobić to samemu.

Jeśli czytając powyższy akapit pomyśleliście: „Zaraz zaraz, trzeba sobie przygotowywać jakieś środowisko? To nie wystarczy odpalić Hollywood w wersji minimum 6.0 i APK Compiler?" to niestety muszę Was rozczarować. Musimy zainstalować i pokonfigurować odpowiednio cały łańcuszek programów i programików. Ale po kolei — czego będziemy potrzebować?

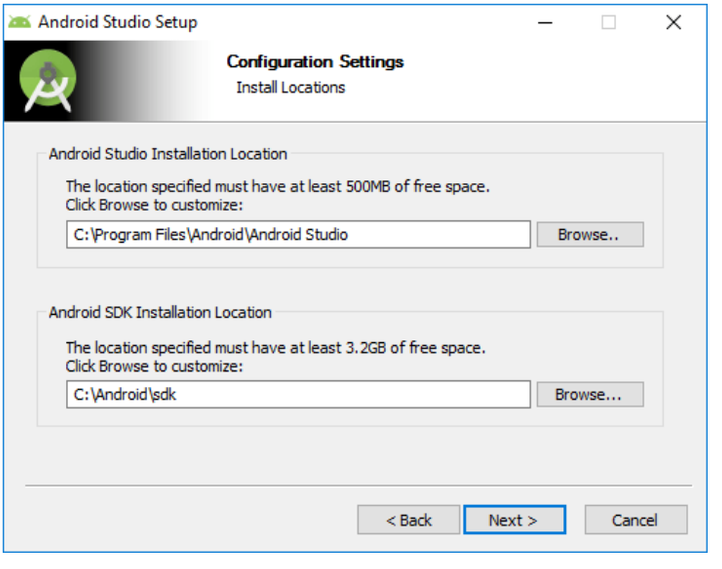
Po pierwsze – 64-bitowego komputera działającego pod kontrolą systemu Windows. Wiem, że ble, ale narzędzia, których używamy po drodze, są dostępne tylko tam. Tu mała uwaga a propos tychże narzędzi — jest to niezmiernie istotne, żeby nazwa użytkownika na wspomnianej Win-maszynie nie zawierała polskich znaczków diakrytycznych. Inaczej proces wywali się po drodze. Ja (ku swej zgryzocie) próbowałem najpierw na koncie o nazwie „Użyszkodnik". Zmiana nazwy nic nie dała, bo ścieżki w systemie i tak zawierały już to nieszczęsne „ż". Pomogło dopiero dodanie nowego konta użytkownika i instalacja wszystkiego „na niego".

Po drugie — trzeba zainstalować tak zwany „Java SE Development Kit" ze strony Oracle.
Z tym nie powinno być żadnych problemów (a przynajmniej nie było ich w moim przypadku).

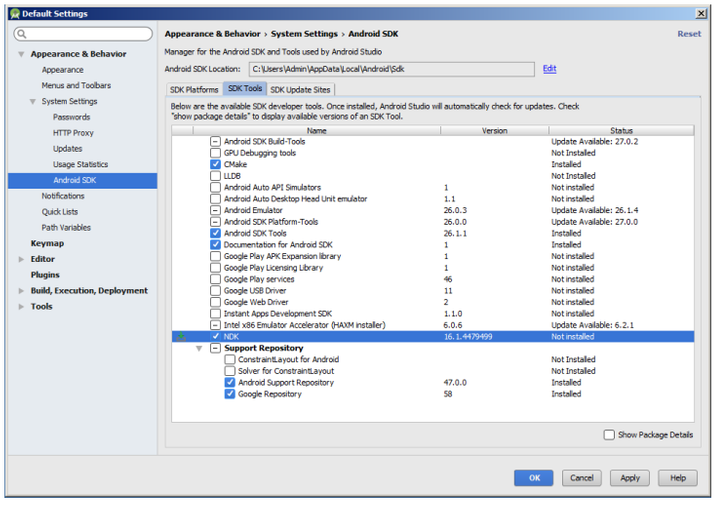


Instalacja "Java SE Development kit"

Trzeci krok to pobranie i instalacja „Android Studio" (w wersji przynajmniej 2.3.1). Zainstaluje nam ono też od razu wymagane SDK. Wypada zapamiętać/wrzucić w wycinek ścieżkę, w jakiej owo SDK będzie zainstalowane, co przyda nam się później w konfiguracji samego Compilera.



Po czwarte — odpalamy dopiero co zainstalowane Android Studio, mlaskamy w „Configure” i nakazujemy doinstalować narzędzia „Cmake” i „NDK” z zakładki „SDK Tools”.



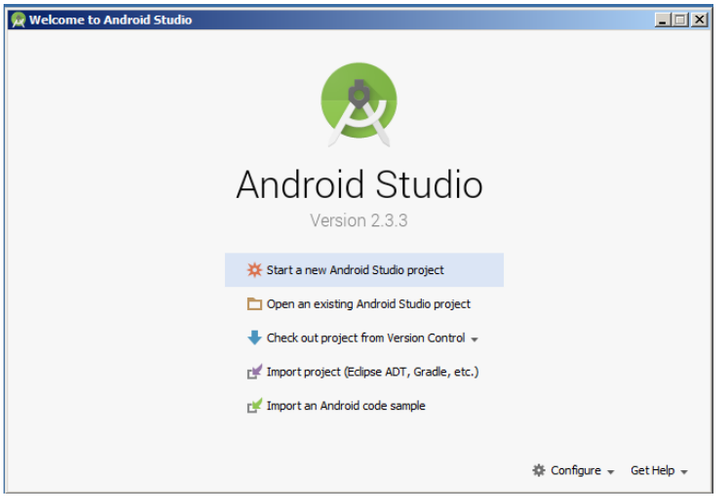
Po tym krwawym trudzie przychodzi pora na instalację narzędzia „Gradle” ze strony gradle.org (wystarczy plik wykonywalny). Zawartość wypakowujemy do lokalizacji C:\Gradle. Tu mała uwaga — kiedy będziemy budować naszą pierwszą APKę, Gradle połączy się z internetem, żeby zainstalować Android Plugin. Z tego powodu „nasz pierwszy raz” (i tylko ten jeden) będziemy musieli wykonać z włączoną siecią.

Przyszła pora na krok szósty — wygenerowanie keystore/klucza do podpisywania naszej apki. Bez tego sklep Google Play nas opluje. W tym celu używamy narzędzia „keytool” (wchodzi ono w skład JDK, które zainstalowaliśmy w punkcie 2). Otwieramy konsolę (koniecznie w trybie administratora) i w folderze „bin” naszego JDK wpisujemy następującą komendę:

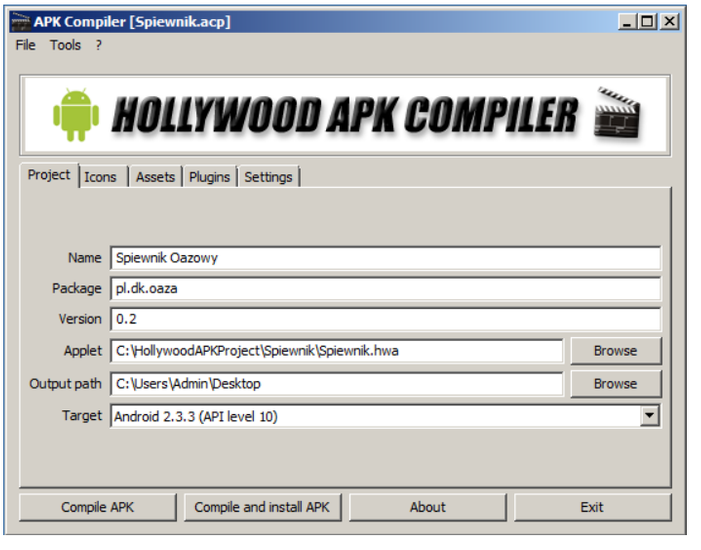
```
keytool -genkey -v keystore <nazwakeystore>.keystore -alias <naszalias> -keyalg RSA -keysize 2048 -validity 10000
```

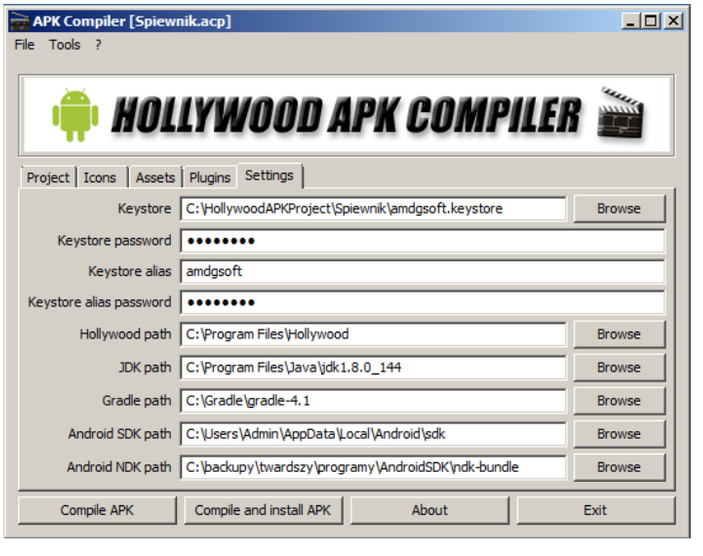
Oczywiście za <nazwakeystore> wpisujemy swoją ulubioną nazwę, a za <naszalias> swoje (swojej firmy) dane, na przykład amigarulez.inc (małymi literami, bo jak użyjemy dużych, albo spacji to możemy mieć problemy). Dociekliwe narzędzie zapyta nas jeszcze o dane osobowe, nazwę firmy, adres, oraz hasło dla keystore i klucza a następnie wygeneruje co trzeba.

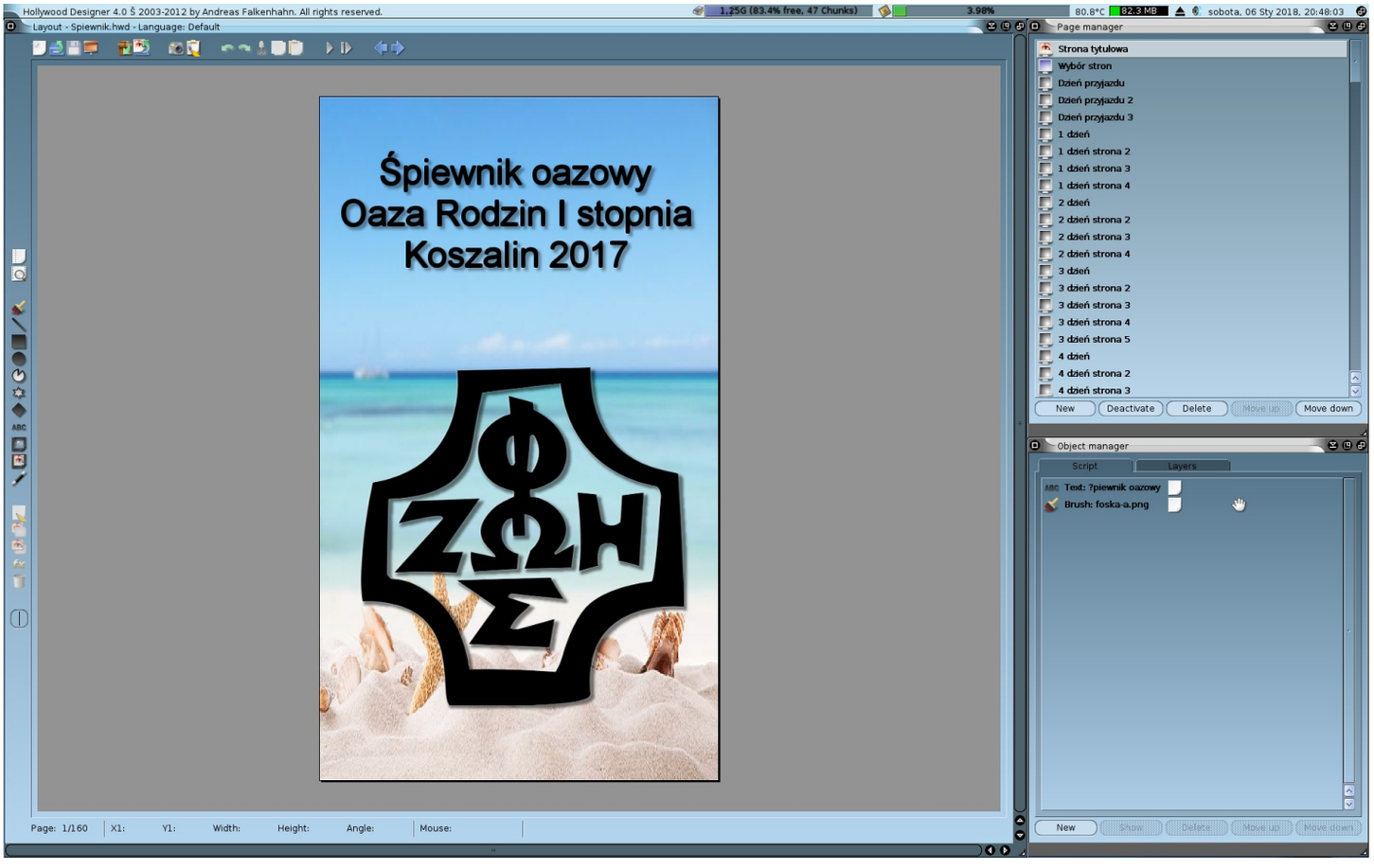
Kolejnym (na szczęście już ostatnim) etapem na naszej drodze do sławy i bogactwa jest uruchomienie Hollywood



APK Compilera i ustawienie prawidłowych ścieżek dla zainstalowanych komponentów. Po ich ustawieniu proponuję projekt zapisać, bo nic tak nie stresuje, jak konieczność ustawiania ich za każdym razem na nowo, gdy musimy wykonać nowy build.





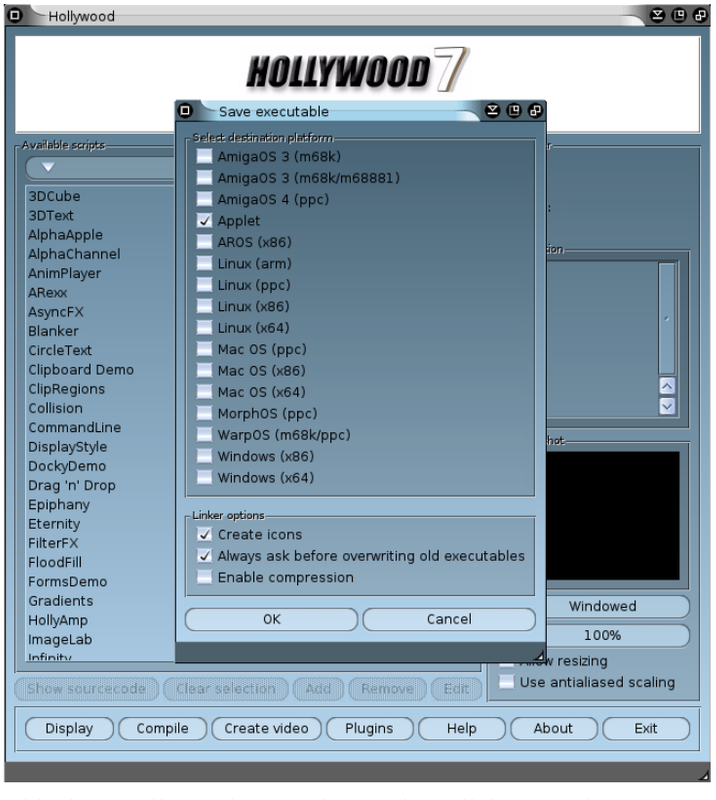


Projekt otwarty w Designerze

Jak stworzyć APK-ę nie umiejąc kompletnie nic, czyli przy użyciu Designera?
To proste jak drut kolczasty. Nasz projekt .hwd zapisujemy jako skrypt „Save script", potem pod Hollywood GUI każemy z tego skryptu skompilować aplet. Jego z kolei przenosimy na PC i polecamy APK Compilerowi zrobić z niego APK, uruchamiamy go na swoim fonie... a potem się dziwimy skąd te błędy i niedociągnięcia.
Ale po kolei:

Użyte pliki zewnętrzne (np. obrazki) oraz czcionki wypada zlinkować z naszym apletem, ewentualnie zrobić to jako „Android assets" z poziomu APK Compilera. Bez tego apka się wywali nie mogąc znaleźć pierwszego lepszego pliku/czcionki.

Kolejna sprawa to tryb wyświetlania — Designer daje nam możliwość ustawienia „Mode" na „Full screen (Auto)", ale lepiej zmienić go ręcznie pod edytorem na FullScreenScale ("auto" w moim przypadku zostawiał dookoła obrazu czarną ramkę).



Okienko z Hollywood GUI w którym kompilujemy applet

Ostatnia uwaga dotyczy polskich znaczków — póki nie ruszamy skryptu pod PC będzie dobrze. Jeśli spróbujemy skrypt wczytać do IDE i przerobić na aplet na pececie, będzie trzeba poprawić wszystkie polskie znaki diakrytyczne ręcznie.

Teraz pozostaje mi Wam już tylko życzyć udanych buildów i milionów pobrań. Do zobaczenia w następnej szkółce!

Rezultat – odpalona nasza „apka” na smartfonie z Androidem

REKLAMA

Dyski NTFS i AROS

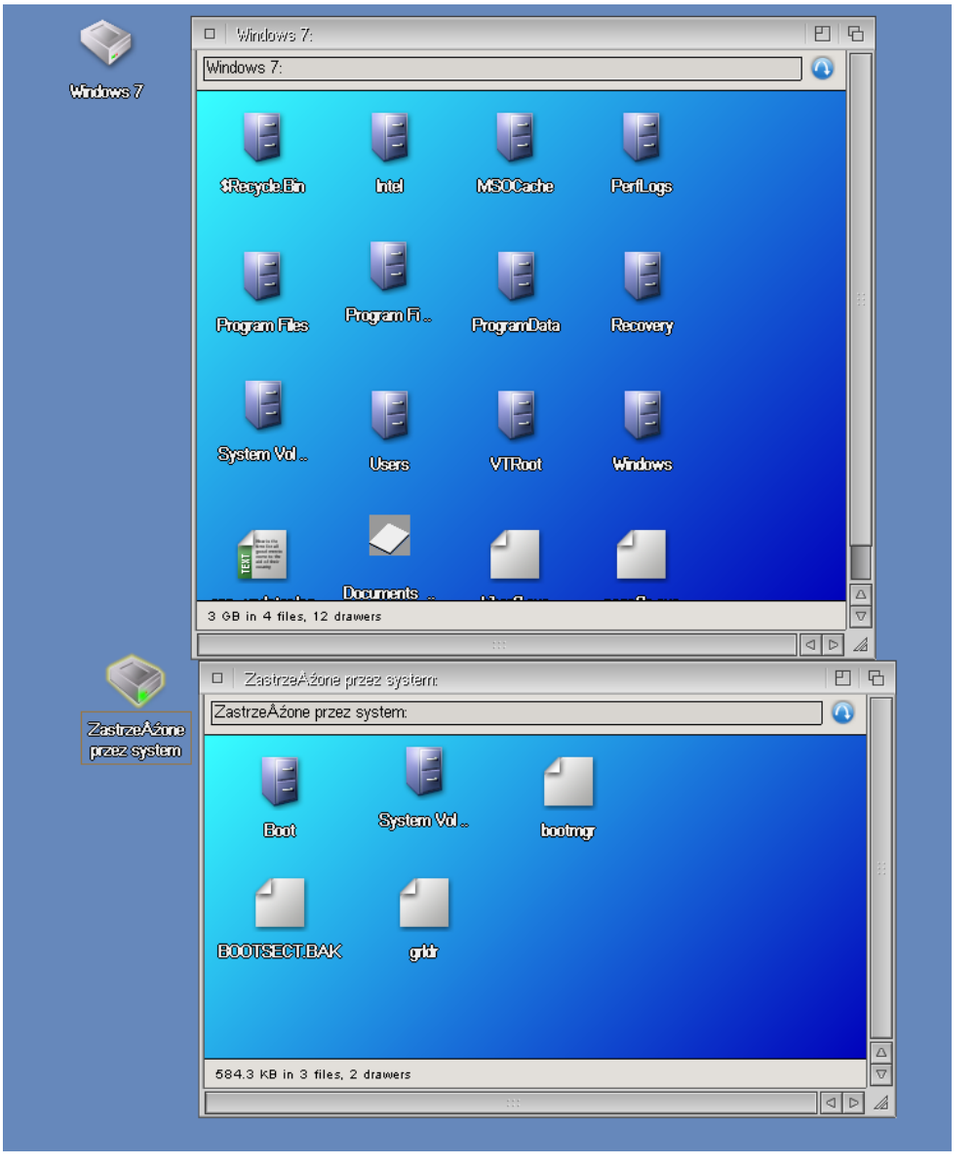
Don Rafito

Jak coś jest do wszystkiego, to jest do niczego, jak rzecze stare porzekadło. Niemniej jednak, multiplatformowość jest domeną naszych czasów. Jednoczesny dostęp do wielu tych samych zasobów z poziomu różnych rozwiązań technologicznych stało się na chwilę obecną już wymaganym standardem.

Podobnie jest z systemem AROS. Bazujący na współczesnych komponentach postamigowy system, dzięki twórcom, próbuje integrować się nie tylko urządzeniami stricte przypisanymi jemu, ale także z dodatkowymi peryferiami; dyskami, pamięcią, płytami rozszerzeń etc., wykorzystywanymi przez inne systemy operacyjne, czasem nawet jednocześnie Dobrym przykładem jest tu dystrybucja Icaros Desktop, w której to tryb Host Bridge na tyle integruje się ze systemem gospodarza, że z poziomu AROS-a, uruchomionego w maszynie wirtualnej, można odpalić aplikacje w nim zainstalowane. Czy to w systemie Windows, czy Linux.

A co jeśli AROS-a mamy zainstalowanego natywnie, np. na drugim twardzielu i chcemy mieć dostęp do wszystkich napędów w komputerze, żeby np. nie przerzucać grafik, filmów, czy muzyki za pomocą pamięci przenośnych, celem wykorzystania potem, tychże plików na postamigowym systemie? Nic prostszego. Wystarczy podpiąć dyski i partycje, niebędące rozpoznawalne standardowo przez system AROS, poprzez stworzenie odpowiedniej mountlisty. Konkretniej, w tym przypadku zmusimy AROS-a, aby widział dyski sformatowane w systemie NTFS.
Do dzieła.

Ze strony AROS Archives (archives.aros-exec.org) ścigamy pakiet *ntfs3g-handler*, autorstwa Fredrika Wikstrom'a, i rozpakowujemy w dowolnym miejscu. Do *Sys: L* kopiujemy plik *ntfs3g-handler*. Następnie otwieramy dowolny edytor tekstu i tworzymy mountlistę:



```
FileSystem = ntfs3g-handler
DosType = 0x4e544653
Device = ata.device
Unit = 0
BlockSize = 512
Surfaces = 1
BlocksPerTrack = 1
LowCyl = 0
HighCyl = 976773166
Reserved = 0
/* Control = ro */
Activate = 1
```

Zapisujemy, tworzymy do pliku ikonkę projektu i podpinamy w tooltypes pod *C: Mount*.
Kolejna sprawa; znowu otwieramy dowolny edytor tekstu i edytujemy plik *L: automount-config* tak, aby zawierał następującą treść:

```
FAT\0 = L: fat-handler
NTFS = L: ntfs3g-handler
```

Zapisujemy. Restartujemy system. Naszym oczom na oknie Wanderera, pojawią się nowe ikony dysków, dysków Windowsa, lub Linuxa, sformatowanych systemem NTFS. Teraz dowolnie możemy otwierać i edytować różne pliki zawarte na nich, przez arosowe aplikacje; muzykę, filmy czy grafikę.
Owa recepta dotyczy dysków podpiętych w trybie SATA w komputerze stacjonarnym. Jeśli mamy zamiar używać dysków przenośnych podpinanych przez port USB, mountlista powinna wyglądać tak:

```
FileSystem = ntfs3g-handler
DosType = 0x4e544653
Device = usbscsi.device
Unit = 0
BlockSize = 512
Surfaces = 1
BlocksPerTrack = 1
LowCyl = 0
HighCyl = 976773166
Reserved = 0
/* Control = ro */
Activate = 1
```

Reszta jak wyżej. Powodzenia.

The Legend of Edgar

Don Rafito

Jest pewna reklama, a właściwie nie reklama, tylko jej końcowa pointa; ta o najszybciej rozwijającej się sieci. Pięknych, wzniosłych, nic nie wnoszących, mających tylko za zadanie naciągnąć klienta na pieniądze haseł, powstało całe mnóstwo. Wyżej wymieniony slogan spokojnie można by zatrudnić do zareklamowania retrogamingu, który prócz pełnych realizmu, zatracających gracza w cyfrowym świecie, powodując jego częściowe i czasowe (przeważnie) odmóżdżenie, gier trójwymiarowych, jest obecnie najszybciej rozwijającą się i zyskującą popularność, linią produkcji oprogramowania służącemu rozrywce. Tylko czy rzeczywiście potrzebna jest tu jakaś reklama? Czy potrzebny był w ogóle ten cały zagmatwany wstęp? Czy nie wystarczyłoby stwierdzenie, że retrogaming tworzą ludzie, pełni pasji, nostalgii i wspomnień, którzy wychowywali się na pikselach w latach 80. i 90. przeżywszy ten skok technologiczny, a teraz są dorośli, dojrzali, czasem okraszeni już lekką siwizną i zasiadają do tych swoich, wymuszonych przez standardy, maszyn wypchanych gigabajtami i gigahercami, tylko po to, żeby zrobić dobry, pełny emocji i napięcia pikselowy tytuł, po który sięgną tacy sami jak oni, a przy okazji młodsi się czegoś nauczą? Wystarczyłoby.

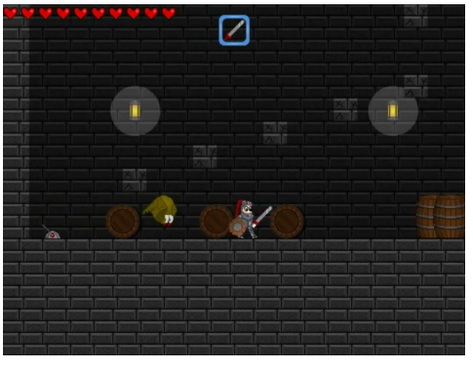
Legenda Edgara lub jak kto woli The *Legend of Edgar*, to kolejna platformówka od **Parallel Realities**, która po tegorocznym update'cie, znowu przypomniała graczom o swoim istnieniu. Tylko że platformówka, to za mało powiedziane. Pozycja ma też mocno zaakcentowany charakter przygodowy. Gracz nie omieszka także posmakowania klimatów gier logicznych i zręcznościowych.
Gdy ojciec tytułowego bohatera, po ciemnej i burzliwej nocy nie wraca do domu, Edgar obawia się najgorszego: jego tato został porwany przez złego czarnoksiężnika, zamieszkującego fortecę za zakazanym bagnem.

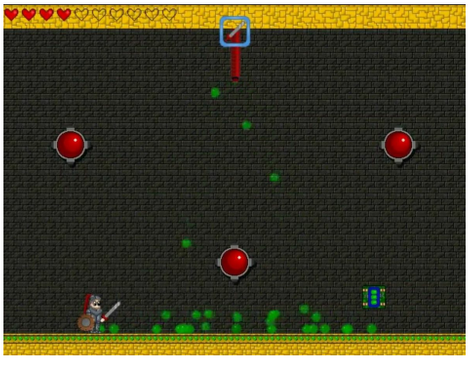
Młodemu nie pozostaje zatem nic innego jak włożyć zbroję i uratować ojczulka. Niestety jak można by było się domyślić, nie będzie to łatwe zadanie.

Nasz bohater do przemierzenia będzie miał wiele poziomów, będzie musiał pokonać wiele stworzeń łącznie z ich bossami, będzie musiał rozwiązywać najróżniejsze łamigłówki, czy szukać różnych tajnych przejść.

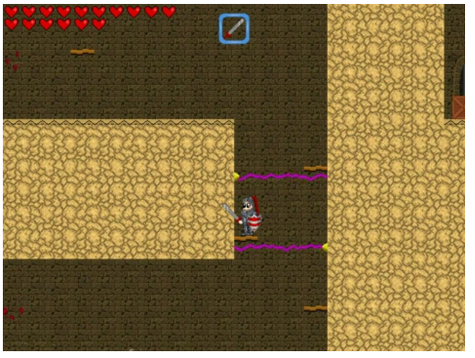
Zbroja, którą włożył na siebie podczas rozpoczynania swojej wędrówki, to nie jedyny oręż. Młody Edgar wraz z przemierzaniem kolejnych światów, zdobywać będzie różne narzędzia machiny wojennej; miecze, łuki, tarcze. Czasem, żeby je pozyskać, przyjdzie mu to odpracować, wchodząc w kolaborację z napotkanymi postaciami. Bardzo przydatne będzie także kolekcjonowanie serduszek, które zapewnią mu dalszy byt podczas przygody.

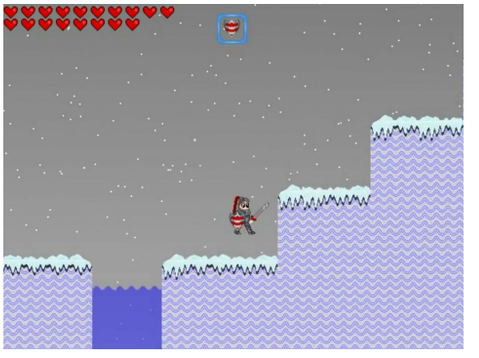
Kiedy pierwszy raz odpaliłem tytuł, na myśl przyszły mi rysunki zrobione



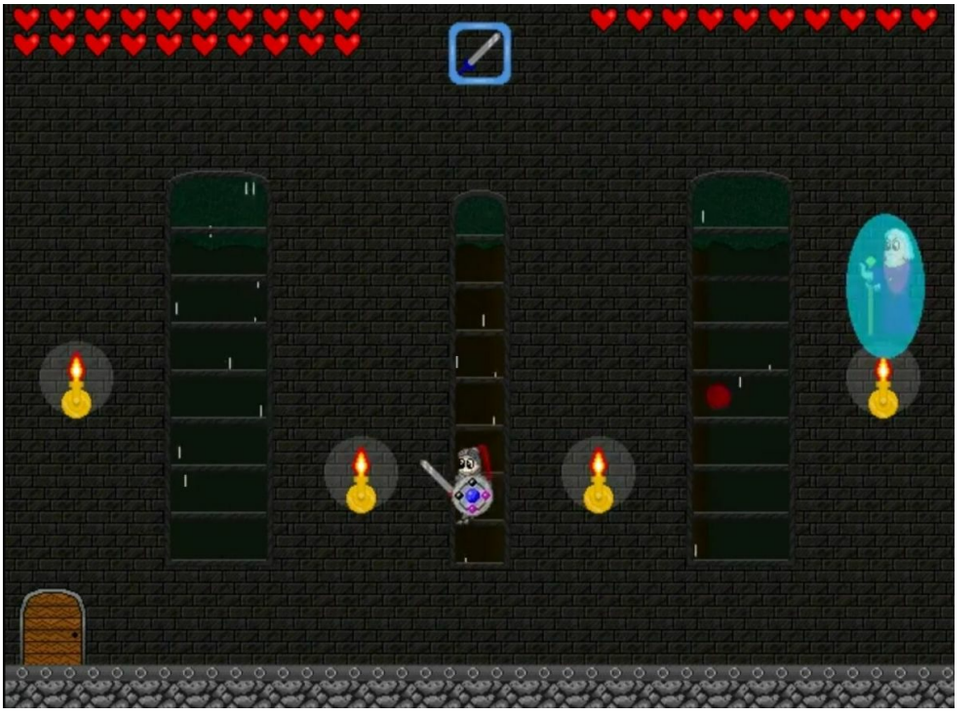


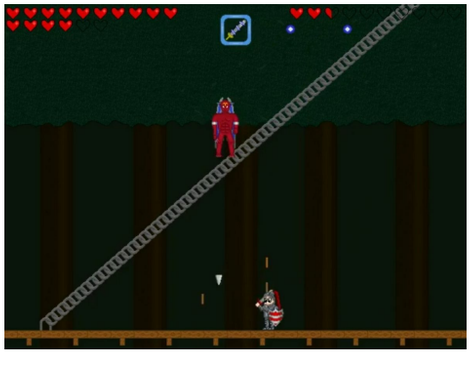
w Deluxe Paintcie. I rzeczywiście tak trochę jest. Szata graficzna jest właśnie taka rysunkowa, ale dość szczegółowa i zróżnicowana. Postacie, elementy scenerii, ba całe poziomy są urozmaicone i naprawdę dobrze wykonane. Animacja postaci, efekty specjalne, wszystko dobrze się prezentuje.





Cała masa bardzo dobrze zrobionych sampli, świetne podsycające klimat danego poziomu podkłady muzyczne, efekty dźwiękowe, wręcz krzyczą soczystością i dokładnością z głośników. Jest tak dobrze, że aż za dobrze jak na grę utrzymaną w klimacie retro. *The Legends of Edgar*, to tytuł, który rozwijany jest już od kilku dobrych



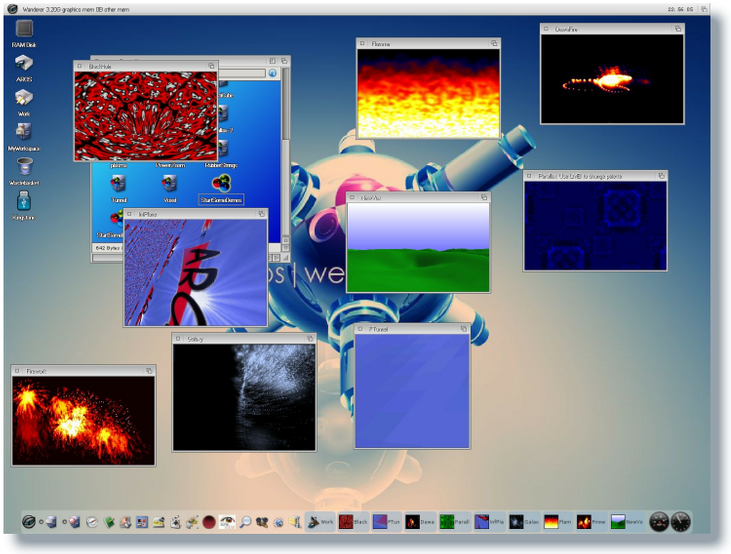


lat. Od czasu do czasu pojawiają się jakieś uaktualnienia i poprawki, dodające grze jakieś nowe korzyści czy likwidujące zauważone błędy.

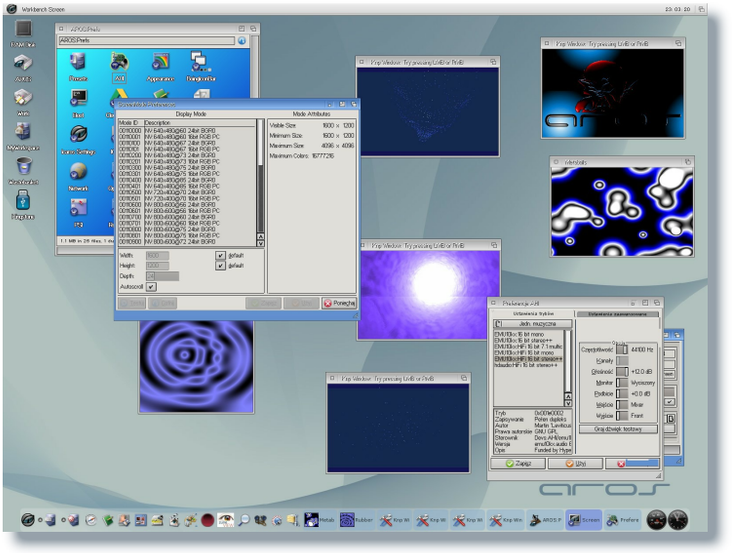
Już dawno nie widziałem tak dobrze wykonanej i dopracowanej pozycji retro. Rozbudowana fabuła, wielowątkowość przygód głównego bohatera, spora możliwość konfiguracji w tym obsługa joypada, co czasem bywa kłopotliwe, jeśli chodzi o gry na AROS-a, porządnie wykonana strona graficzna i dźwiękowa. Do tego niebagatelny, w miarę prowadzania rozgrywki rosnący, poziom trudności. To wszystko sprawia, że tytuł zostanie doceniony i z satysfakcją skonsumowany nie tylko przez wymagających retrograczy, ale także wszystkich tych, którzy szukają w grach czegoś więcej niż strzelania, zabijania i wysadzania. Polecam.

Tytuł: The Legend of Edgar
Platforma: AROS i386, starsze wersje AOS4.x i MOS
Firma: Parallel Realities
Rok produkcji: 2011-2017
Grafika: 9
Muzyka: 9
Grywalność:9

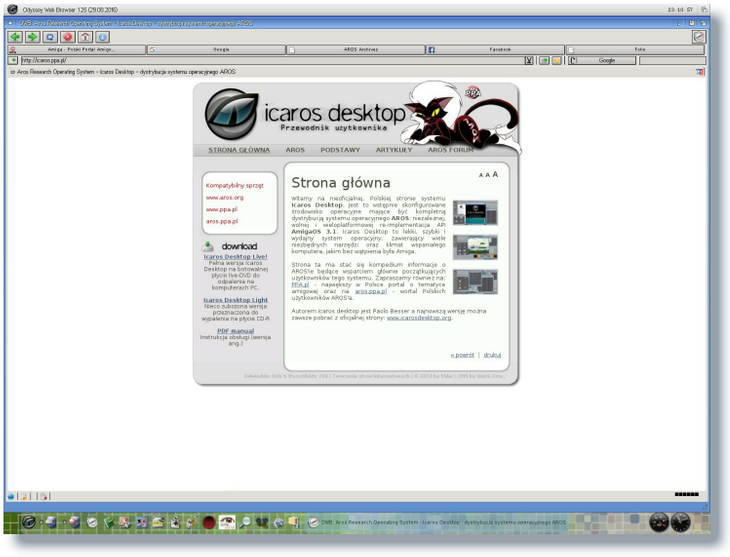
Galeria AROS-a



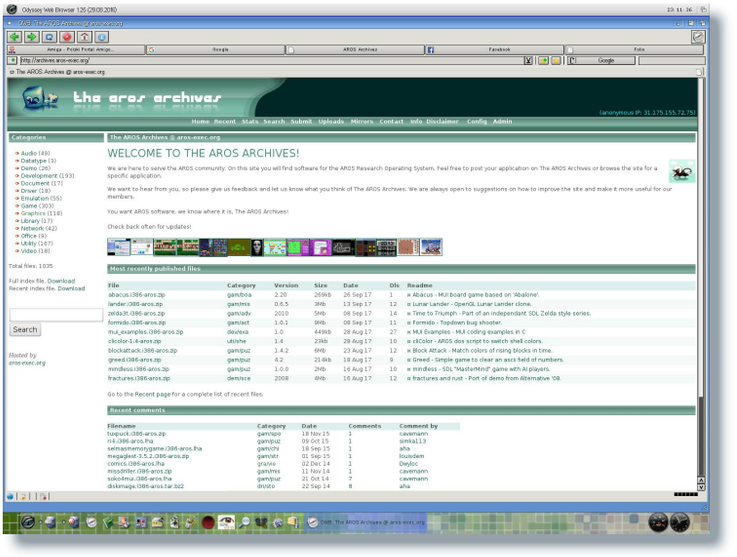
Icaros Desktop – demka wyświetlane współbieżnie



Pokazane preferencje wyświetlania i AHI



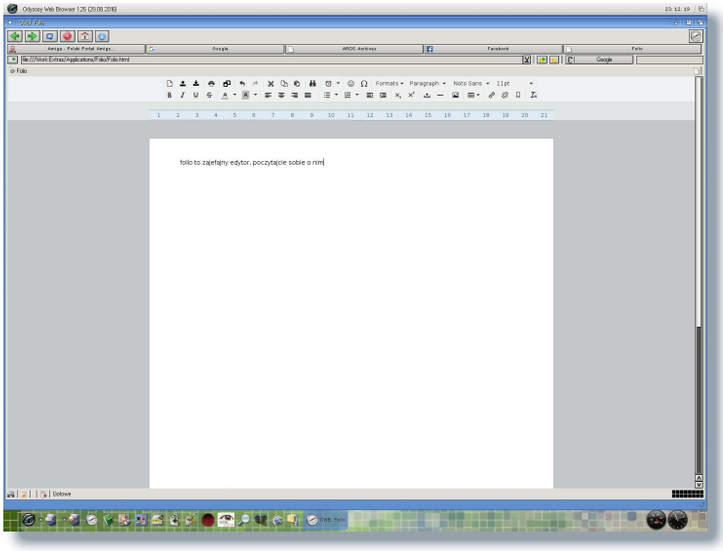
Polska strona o AROS-ie – http://icaros.ppa.pl



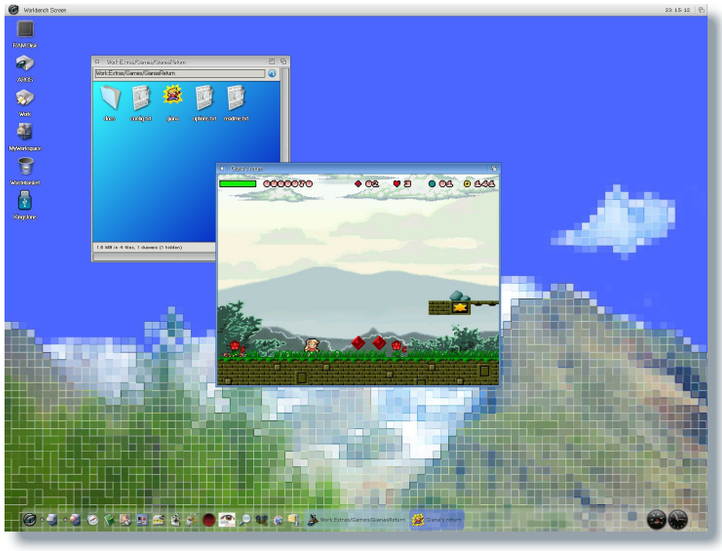
The AROS archives – http://archives.aros-exec.org

Podobno „jeden obraz wart więcej niż tysiąc słów”. Zrzuty prezentujące AROS-a na swoim komputerze nadesłał Don Rafito. Mam nadzieję że spodoba się wam taka forma prezentacji systemów.

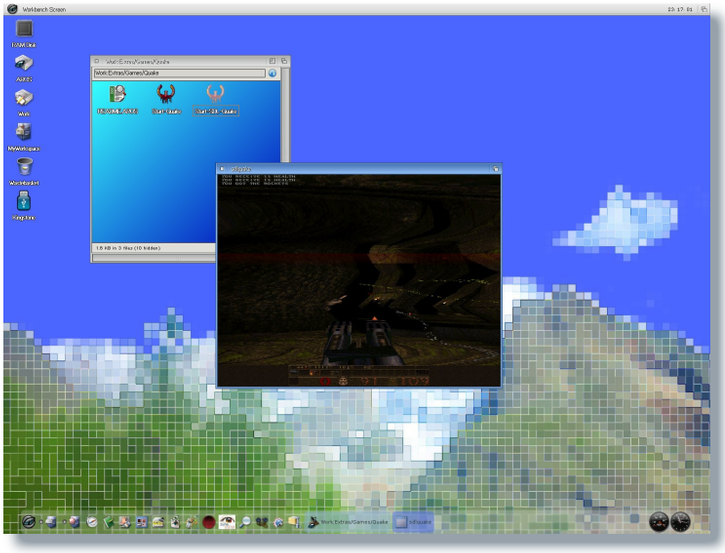
Maskotka systemu – Kitty.
Jeżeli komuś kojarzy się ona z Sabriną z serii krótkich komiksów – to dobrze skojarzył. Autor jest ten sam – znany amigowy rysownik Eric W. Schwartz.



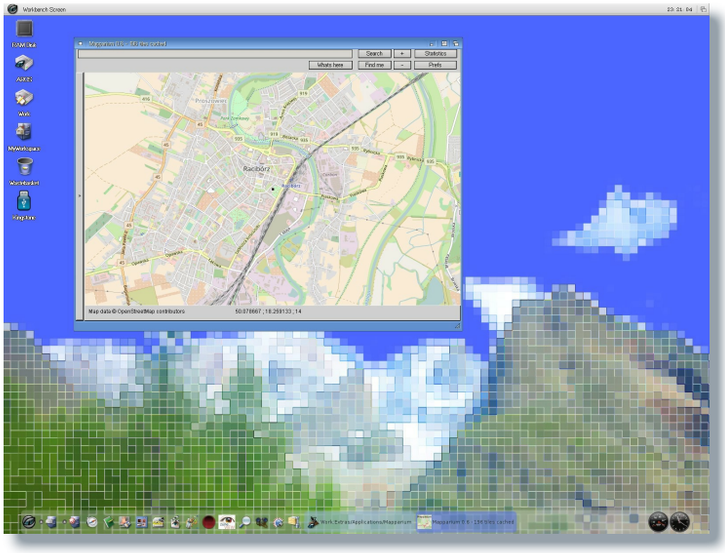
Przeglądarkowy edytor tekstu *Folio*, nie wymaga połączenia online



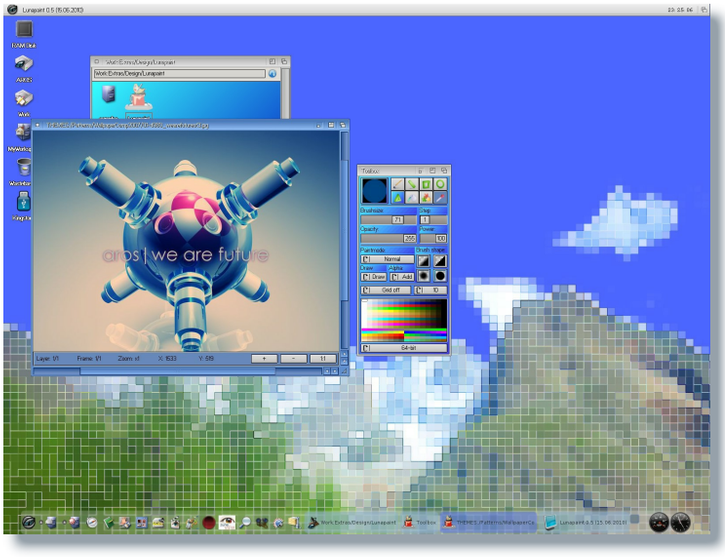
Gra platformowa *Giana's Return*



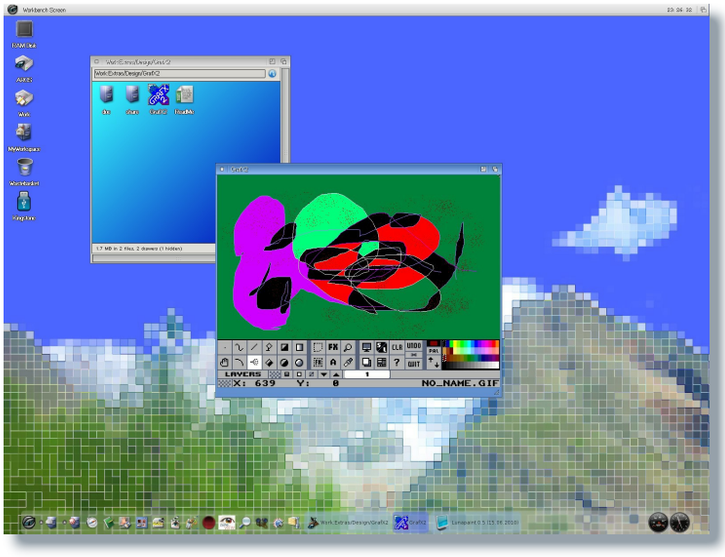
Quake



Mapparium, recenzja w poprzednim numerze naszego czasopisma



Lunapaint, program graficzny



GrafX2, inspirowany Deluxe Paintem i Brilliancem

Powrót do Amigi po latach – moje doświadczenia z NG

jarokuczi

Wszystko zaczęło się w okolicach 1995 roku. Wtedy to udało się rodziców namówić na przesiadkę z wysłużonego C64 na Amigę. Wybór modelu niestety na tym etapie był poza moją kompetencją. Rodzice mogli sobie pozwolić na poczciwą A600. Tak zaczęła się wspaniała przygoda, która uczyniła ze mnie taką osobę, jaką jestem teraz. Gdy okazało się, że A600 nie da się, w tamtych czasach, sensownie rozbudować a ja chciałem od komputera czegoś więcej niż wczytywanie gier z dyskietek. Zapadła decyzja o wymianie z dopłatą na A1200. Później to było już z górki aż do Blizzarda PPC i BVision w obudowie tower. Niestety ta konfiguracja starczyła (wydajnościowo do mojej pracy, którą i tak wykonywałem głównie pod linuxem) mniej więcej do roku 2005. Był to czas, w którym Amiga poszła do szafy, a na biurku zagościł klon.

Amigę ceniłem głównie za wspaniały system operacyjny, który nie był prawie rozwijany w czasie, w którym używałem Amigi. Przez kolejne lata zmieniałem komputery na bardziej wydajne. Z uwagą śledziłem Ami światek z boku, zaglądając na portale PPA i EXEC. Pojawił się Pegasos z MorphOS-em, Sam440ep z AmigaOS 4 oraz AROS. No dobra ten ostatni pojawił się zdecydowanie wcześniej, ale wtedy zaczynał powoli się do czegoś nadawać. Zawsze chciałem zobaczyć któryś z tych systemów w akcji, niestety przez długi czas żadnego, poza AROS-em nie było mi dane.

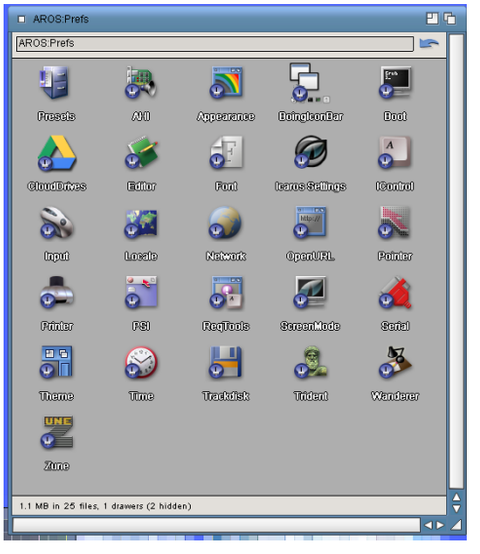
W tym artykule opiszę każdy z trzech systemów NG okiem kogoś, kto zatrzymał się na AmigaOS 3.9, który to będzie moim punktem odniesienia.

AROS

Jako że AROS to projekt z otwartymi źródłami i w początkowych fazach projektu działający na platformie x86 z uwagą śledziłem jego rozwój od samego początku (dla mnie była to wzmianka w którymś z numerów Magazynu Amiga). Niestety przez długi czas mój komputer nie był na tyle wydajny żeby sensownie wirtualizować sobie maszynę z AROS-em, instalacja, jako drugi system operacyjny nie wchodziła w grę. Dopiero przy kolejnej zmianie sprzętu postanowiłem sobie zostawić starego grzyba do zabawy z AROS-em.

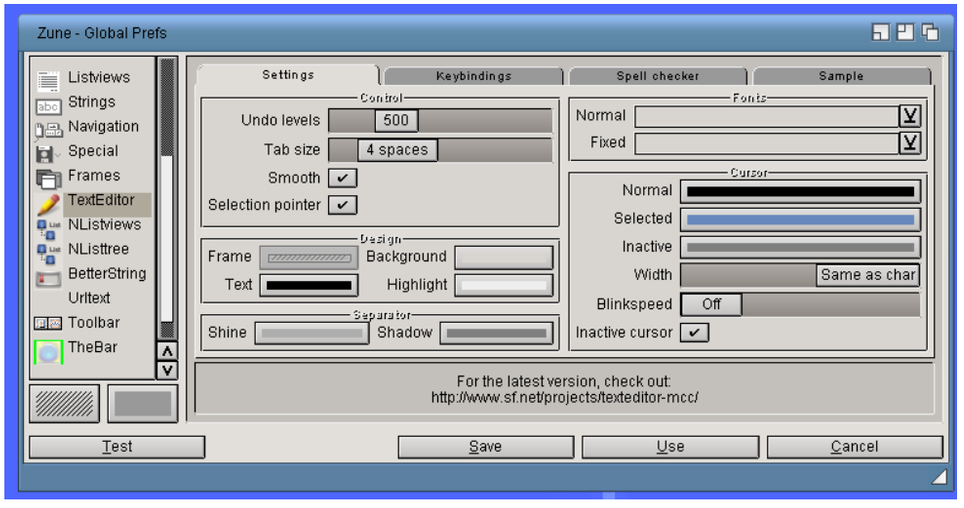
Na pierwszy i zarazem ostatni ogień poszedł AROS w dystrybucji Icaros Desktop (dawniej vmwaros). To, co ukazało się moim oczom zrobiło niesamowite wrażenie. Bardzo duża baza przeróżnego oprogramowania dostępna zaraz po zainstalowaniu. System, który nie „straszy” 4 kolorami na starcie był dla mnie niesamowitym skokiem jakościowym, takie wejście w XXI wiek.

Sam system ze swoim menadżerem plików Wanderer, oraz strukturą dysku systemowego zgodnego z AmigaOS wydawał się raczej ewolucją niż rewolucją. Ten fakt wynikał z tego, że w założeniach miał to być system zgodny z AmigaOS 3.1 na poziomie API.



Okienko Wanderera pokazującego zawartość katalogu Prefs

Wanderer – menadżer plików, to trochę bardziej rozbudowany klon Workbencha. Główną różnicą, rzucającą się na pierwszy rzut oka to pasek narzędziowy, zawierający pole ze ścieżką katalogu oraz przyciskiem wstecz, oraz pasek statusu na dole okna z informacjami o zawartości katalogu. Dzięki przyciskowi wstecz po katalogach możemy się poruszać jak w mainstreamowych systemach operacyjnych nie otwierając nowego okna dla każdego podkatalogu. To domyślne zachowanie możemy zmienić na takie jak w Workbenchu.



Preferencje ZUNE, zupełnie jak MUI

Zune (MUI) to otwartoźródłowy zamiennik dla MUI. Podobnie jak sam AROS w stosunku do pierwowzoru MUI jest/powinien być kompatybilny na poziomie API z MUI. Czego efektem jest sporo klas MUI kompilowanych często równolegle dla AmigaOS, AROS i MorphOS. Wiele aplikacji, które na AmigaOS wymagają MUI kompilowana, jest dla AROS-a z wykorzystaniem ZUNE, np. YAM, SimpleMail.

Prefs to katalog na dysku systemowym, w którym w AmigaOS mamy umiejscowione programy zmieniające ustawienia systemu jak i komponentów zainstalowanych przez użytkownika. W AROS-ie sprawa ma się dokładnie tak samo. Z tą różnicą, że nie wszystkie systemowe programy preferencyjne mają swoje odpowiedniki w oryginale.

Kompatybilność z oprogramowaniem dla AmigaOS m68k, w odróżnieniu od pozostałych systemów NG posiadających możliwość uruchamiania takich programów „natywnie”, możemy uzyskać jedynie poprzez emulację całego komputera. Co prawda jest integracja z emulatorem, która umożliwia odpalanie aplikacji w poczuciu, że są prawie natywne, jednak to tylko emulacja całego komputera. Aplikacje tak uruchomione nie mogą rozmawiać z API systemowym.

Problemy ze sprzętem to niestety bardzo poważny kłopot w świecie x86 gdzie chipsetów, kart sieciowych czy muzycznych jak mrówek. W moim przypadku ostatecznie zainwestowałem w poleasingowy komputer Fujitsu-Siemens z procesorem Core 2 Duo, wspieraną kartą dźwiękową Realtek, i dodatkowo dokupiłem kartę graficzną GeForce 9500 GT. Komputer ten pracuje też jako dedykowana maszyna do emulacji Amigi.

Podsumowując AROS to całkiem fajne przedsięwzięcie, które od jakiegoś czasu, jako jedyny system NG jest dostępne na platformę m68k. Niestety, jako twór, który rozwija się własnym życiem bez jakiegokolwiek przywództwa, w wielu płaszczyznach jest niedopracowany. Jednak, gdy patrzę na to jak się zmienił od ostatnich dziesięciu lat, widzę bardzo dużą poprawę stabilności.

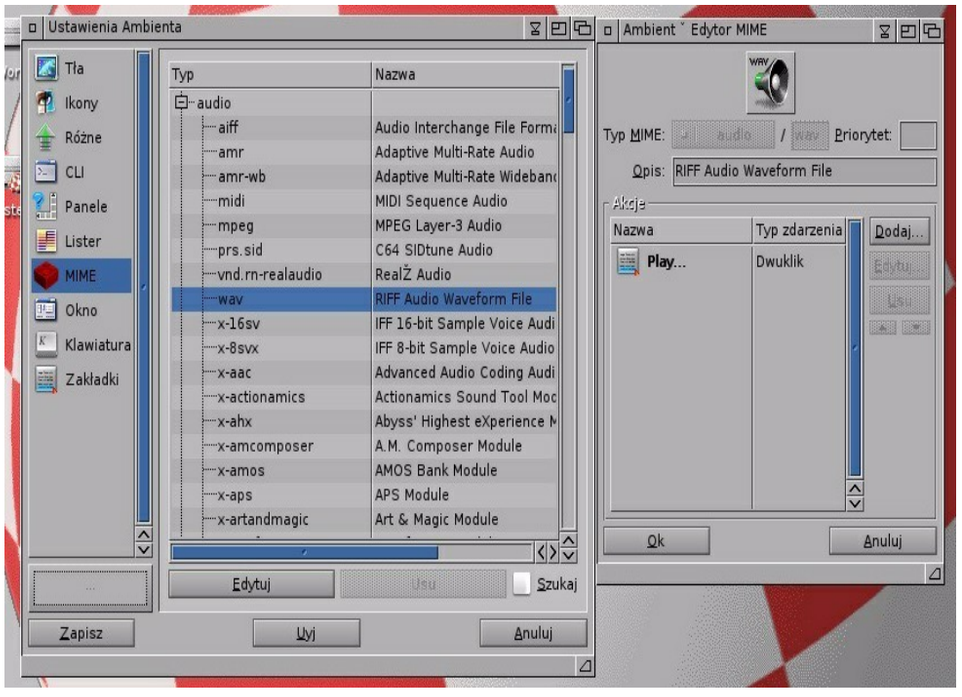
MORPHOS

Ten system zagościł u mnie w domu w momencie, gdy stary sprzęt Apple zaczął być tani jak przysłowiowy barszcz. Za 12” iBooka G4 800 MHz zapłaciłem 80 zł + koszty przesyłki, a za Power Maca G5 DP 1,8 GHz 70 zł + koszty przesyłki. Muszę przyznać, że kosztowy próg wejścia w świat NG od strony Niebieskiego Motyla obecnie jest praktycznie zerowy. Pobieranie i używanie systemu nic nie kosztuje, jeśli godzimy się z jego ograniczeniem w wersji niezarejestrowanej (ja wytrzymałem około roku). Czyli makabrycznego zwolnienia komputera po 30 minutach pracy, na które lekarstwem jest reset.

Do tego systemu podchodziłem już mając za sobą sporo doświadczeń w zabawie z AROS-em. Wydawało mi się, że nic mnie nie zaskoczy. Miałem rację, wydawało mi się. Ograniczenie możliwości uruchamiania systemu na kilku z góry określonych modelach sprzętu zaowocowało tym, że każdy sprzęt w iBooku zadziałał jak należy bez zbędnych kombinacji. MorphOS wydał się dla mnie w wielu aspektach rewolucyjny. Całość UI jest zbudowana w oparciu o MUI w związku, z czym każdy aspekt systemu można konfigurować wedle własnego widzimisię.

Belka ekranowa w końcu potrafi nie tylko wyświetlać tekst, ale dodatkowe moduły/widżety:

Jego menadżer plików Ambient czerpie garściami z funkcji dostępnych w menadżerach w innych systemach operacyjnych. Traktuje archiwa jak podkatalogi. Automatycznie sortuje zawartość katalogów (szok!!!). Zawiera ładny edytowalny pasek narzędziowy. Obsługa ikon



Ustawienia MIME/Def Icons w Ambiencie

domyślnych odbywa się na zupełnie innych zasadach niż w pierwowzorze. Ambient, poza funkcją przeglądania plików, daje możliwość tworzenia paneli, które są odpowiednikiem AmiDocka. Niestety brakuje mu funkcji uruchomienia programu poprzez upuszczenie nań pliku.
W odróżnieniu od AmigaOS i AROS-a, MorphOS nie posiada wielu programów do zmiany preferencji systemowych. Zamiast tego jest jeden program, który przypomina systemowe ustawienia z macOS/OS X.

Poza różnicami wizualnymi, sporo różnic mamy pod maską. Zauważalną na pierwszy rzut oka różnicą jest układ partycji systemowej. Oprócz

Preferencje MorphOS

standardowego przypisania SYS: które kieruje na partycję systemową mamy przypisanie MOSSYS: do katalogu MorphOS na partycji z systemem. Struktura katalogów odpowiada tej, którą znamy z klasycznego systemu. W tym katalogu zainstalowany jest prawie cały system, zmiany wprowadzane przez użytkownika w tym dogrywane biblioteki czy komendy w ogóle nie mieszają w systemie, dzięki czemu aktualizacja systemu powinna odbywać się bezboleśnie. Niestety, nie dane mi było sprawdzić, bo odkąd jestem użytkownikiem MorphOS mamy wersję 3.9. System ten wydał mi się najbardziej dopracowany i rewolucyjny ze wszystkich tu omawianych. Dla mnie jest obecnie numerem jeden w NG świecie.

AMIGAOS 4.1

Na koniec zostawiłem sobie system, który nie tylko wzoruje się i jest kompatybilny z AmigaOS 3.1, lecz taki który jest jego pełnoprawnym następcą. Muszę zastrzec, że systemu tego używałem wyłącznie pod WinUAE. Stąd być może moje opinie są takie jakie są.

Systemu tego spróbowałem kilka miesięcy po tym jak Hyperion ogłosił obniżkę ceny dla edycji classic. System z jego niebiesko-szarą skórką, która wyglądem gadżetów bardzo przypomina poprzednika, wkradł się w moje łaski wiele lat przed tym jak pierwszy raz włączyłem ten system. Przez długi czas używałem poprzednio opisanych systemów (oraz systemu 3.9 z AfaOS) właśnie ze skórkami naśladującymi AmigaOS 4. Sam system, to faktycznie odświeżony AmigaOS 3.1 albo w zasadzie 3.9. Interfejs użytkownika jest zbudowany w oparciu o bibliotekę Reaction, pod każdym wzglę-

dem ustępującą MUI, które i tak jest zainstalowane w systemie bo wiele aplikacji korzysta z tego frameworka. Do dyspozycji dostajemy AmiDock, UnArc, Editpad, oraz wiele innych narzędzi znanych z AmigaOS 3.9 tylko że w wersji PPC.

Workbench to dalej prawie ten sam Workbench co w wersji 3.9. Jako użytkownik tejże wersji poczułem się tu jak w domu. W moim odczuciu sam system nie jest w żadnym stopniu rewolucyjny, poza odświeżonym (bardzo udanie) wyglądem oraz zmianą architektury, jest nie wnoszącą za wiele aktualizacją. Ot kompilacja AmigaOS pod nową architekturę.

Żałuję że nie miałem okazji pracować dłużej na tej wersji zwłaszcza na dedykowanym do tego sprzęcie, być może odkryłbym więcej fajnych właściwości i funkcji. Niestety wysokie ceny wydajnych komputerów kompatybilnych z AmigaOS 4 powodują, że te są poza moim zasięgiem. Używanie go pod emulatorem niestety do najprzyjemniejszych zadań nie należy.
Osobiście uważam, że dużo wygodniej jest użyć w emulatorze systemu w wersji m68k ponieważ emulacja procesora PowerPC jest na tę chwilę strasznie powolna.

Preferencje AmigaOS 4.1

Przeglądarka OWB na AmigaOS 4.1

PODSUMOWANIE

Gdybym obecnie stał przed decyzją, który system NG wybrać, zdecydowałbym się na MorphOS, głównie z powodu niskiej ceny sprzętu (w tym laptopów) kompatybilnego z tym systemem.
AmigaOS 4.1 odpada z dwóch powodów. Pierwszy to nie wina systemu tylko tego, że nie posiadam worka pieniędzy na w miarę wydajny sprzęt z nim kompatybilny. Drugi to brak (póki co) komputera typu laptop na którym można go zainstalować. Ten ostatni jest naciągany bo pewnie odpowiedni laptop łapał by się pod powód nr 1 ze względu na cenę.
Jeśli weźmiemy na tapetę AROS-a to system jest kompatybilny z laptopami i sprzęt można kupić za bezcen, co jest na plus. Niestety system jest niedopracowany i pełny drobniutkich lecz często irytujących błędów, co jest pierwszym minusem. Brak „natywnej" kompatybilności z oprogramowaniem dla AmigaOS m68k oraz braki w natywnym oprogramowaniu czyni system mniej użyteczny niż konkurencja co jest kolejnym minusem.

Bardzo mocno kibicuję AROS-owi i AmigaOS 4.1, ale obecnie stawiam na niebieskich.

Humor

W tym numerze zamiast tekstu z cyklu „Styl życia” jest właśnie „Humor”. Zamieszczamy jeden długi kawał „z brodą”. Niestety dowcipy o amigowcach „NG” jeszcze chyba nie powstały, dlatego miłym przełomem jest odcinek KKA (”Krótki Komiks Amigowy”, znany na PPA), stworzony dla nas przez recedenta.

Gdyby komputery były samolotami...

~~~~~~

Do klienta (nowe linie lotnicze) przychodzi dealer sprzedający Amibusy.

D(ealer): ...jak już mówiłem nasze samoloty są najlepsze. Tańsze od konkurencyjnych DC486, DC586, DC686 nie mówiąc o tych latających trumnach DC-PE...2

K(lient): [słucha]

D: Nasz samolot może lecieć jednocześnie w dwie strony...

K: Niesamowite...

D: Układ sterowania jest najgenialniejszy na świecie. Daje pilotowi tyle możliwości... Swoja droga Pan wie jakie matoły latają na DeCetach...Takiemu to wszystko podkładają pod nos – tylko leci... (tu dealer przełyka ślinę z goryczą). Nie ma pojęcia do jakiej finezji można dojść pilotując Amibusa.

K: Powiedzmy, że kupię Amibusa, czy mogę nim...

[tu dealer przerywa]

D: Jasne że tak!!!

K: ...polecieć do Londynu?

D: Tak! ... Chociaż.. powiedział Pan do Londynu? To dalej niż 50 km? W takim razie będzie trzeba rozszerzyć stateczniki, standardowo montujemy po 1 m, ale jak pan chce lecieć dalej, to konieczne są 2 m.

K: [słucha]

D: Poza tym będzie pan leciał nad wodą? Radzę zaopatrzyć się w stały bak paliwa. Standardowo wyposażamy nasze samoloty w kanistry po 0.8 tys litrów. Rozumie pan... tak jest taniej. Ale trzeba dolewać w czasie lotu... Radzę panu, niech sobie pan zamontuje bak 850 tys litrów. Chociaż może mieć pan kłopoty z kabelkiem paliwowym...

K: Acha...

D: A w ogóle to niech najlepiej pan pożyczy na ten lot DeCeta – one się do takich nudnych zadań nadają. Nie to, co Amibusy... To są maszyny... Prawdziwa finezja... Zwrotne...

K: Mój znajomy ma Amibusa A-500, czy mógłbym od niego pożyczyć trochę paliwa do mojego A-1200?

D: Lepiej nie. To tylko w tych DeCetach do każdego silnika można wlać to samo od DC-XT do DC-PE. Wyobraża pan sobie co to za marnotrawstwo? Po prostu brak ambicji. My stosujemy za każdym razem rozwiązania unikatowe...

K: Dobrze, na koniec jeszcze jedno pytanie, kto produkuje wasze samoloty?

D: Obecnie?

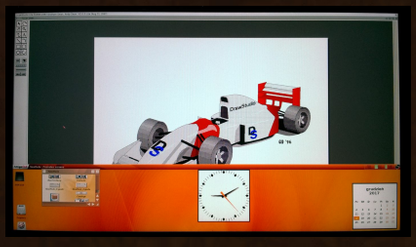
K: Tak

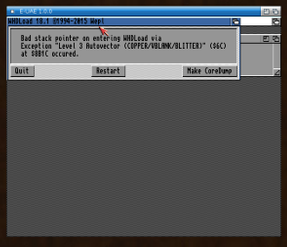
D: Zwykle znajdujemy je na polach z kapustą.
~~~~~~

W następnym numerze

W numerze „2” jest trochę publicystyki, ale cały czas mamy na uwadze popularyzację NG.
Będziemy kłaść nacisk na naukę, na pokazywanie praktycznego użytkowania naszych komputerów.
Niestety proces powstawania naszego magazynu wyklucza coś takiego jak planowanie. Wszystko powstaje w prywatnym czasie każdego z nas, a życie bezwzględnie weryfikuje plany, nawet te rzeczy, które wydają się pewne na 100%. Numer „2” był zapowiedziany jako numer zdominowany przez AROS-a – jak widać nie stało się tak.

Na numer „3” mamy zdeklarowane następujące tematy:



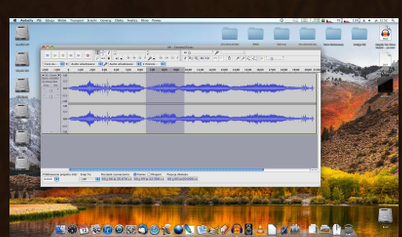


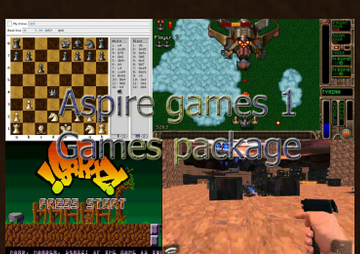
AmigaOS 4.1

- dalszy ciąg szkółki Hollywood
- programy do grafiki wektorowej – kontynuacja zagadnienia; wejdziemy głębiej w temat, praktycznie – czyli jak zacząć w DrawStudio i AmiFIG
- jak poprawić kompatybilność – uruchamianie starszego oprogramowania

MorphOS

- naprawiamy zasilacz do iBooka/Powerbooka G4
- alternatywa dla MorphOS-a w dual-boocie, czyli najlepsze oprogramowanie dla MacOS X





AROS

- Aspire Games 1.1 Game Package – przegląd zawartości
- Hermes – recenzja gry
-

Na pewno będzie więcej treści niż podane wyżej, zwłaszcza że ten numer jest sporo grubszy od poprzedniego (32 -> 58), więc poprzeczka została zawieszona wysoko.

Wyglądajcie numeru „3” w okolicach 15 kwietnia.
Bieżące informacje – jak zwykle na stronie amiga.org.pl

AmigaOne X5000, wersja ze sklepu Amiga On The Lake
(”Boing” nie jest grawerowany, za to czerwony)
Zdjęcie: Aaron Smith